(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 389 175 A2**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 90302749.8

(51) Int. Cl.5: G06F 9/38

(22) Date of filing: 15.03.90

(30) Priority: 15.03.89 JP 62834/89
15.03.89 JP 62832/89
15.03.89 JP 62833/89
16.03.89 JP 62032/89
20.03.89 JP 68377/89
20.03.89 JP 68378/89
26.04.89 JP 108370/89

(43) Date of publication of application:
26.09.90 Bulletin 90/39

(84) Designated Contracting States:
DE FR GB

(71) Applicant: FUJITSU LIMITED
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa 211(JP)

Applicant: FUJITSU MICROCOMPUTER SYSTEMS LIMITED
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa-ken 211(JP)

(72) Inventor: Taniai, Takayoshi
1339, Shimosakunobe, Takatsu-ku
Kawasaki-shi, Kanagawa 213(JP)
Inventor: Sato, Hajime
1-4-12, Kitamagome, Ota-ku
Tokyo 143(JP)
Inventor: Shimura, Hidetoshi
Eri Haitsu A-103, 13-7, Shukugawara 6-chome
Tama-ku, Kawasaki-shi, Kanagawa 214(JP)
Inventor: Saitoh, Tadashi
TOP Shinmaruko 306, 1-1525, Sanno-cho, Kamimaruko
Nakahara-ku, Kawasaki-shi, Kanagawa 211(JP)
Inventor: Oyamada, Shinji
FUJITSU Dalichi Fujigaoki-ryo 211, 23-10, Umegaoka
Midori-ku, Yokohama-shi, Kanagawa 227(JP)

(74) Representative: Stebbing, Timothy Charles et al
Haseltine Lake & Co. Hazlitt House 28
Southampton Buildings Chancery Lane
London WC2A 1AT(GB)

(54) Data prefetch system.

(57) A data processing device (500) contains an execution circuit (200) and a data buffer circuit (210, 213). The above data buffer circuit (210, 213) stores one or more commands and/or one or more parameters which are prefetched, until each of the commands and parameters is read out by the above execution circuit (200). The above execution circuit (200) inputs a command which is the oldest one of the commands stored in the data buffer circuit (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in the data buffer circuit (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein. The above data processing device (500) further contains an insufficient space detecting circuit (60, 80, 231 - 235, 237, 238, 239) for detecting a first state of the data buffer circuit (210, 213) in which state there is not enough vacant space in the data buffer circuit (210, 213) in which a further command and/or a parameter can be stored, and a least data detecting circuit (60, 80, 235, 236, 238) for detecting a second state of the data buffer circuit (210, 213) in which state the data buffer circuit (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in the above execution circuit (200). A prefetch control circuit (100), provided in either in the device (500) or outside of the device, starts the prefetch operation and continues successive prefetch operations when the second state is detected, and stops the prefetch operation when the first state is detected. The system using the above device (500) contains a memory (502)

EP 0 389 175 A2

which stores the above commands and parameters in the order of execution in the execution circuit (200).

Fig. 3A

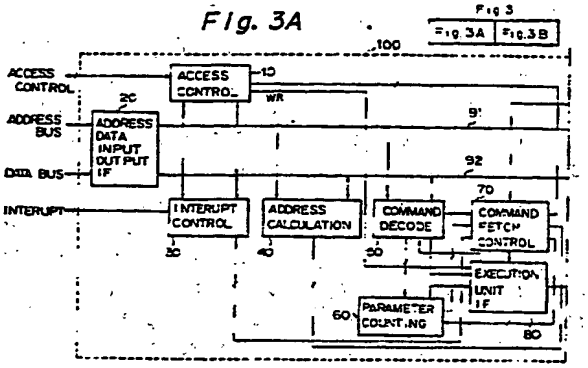
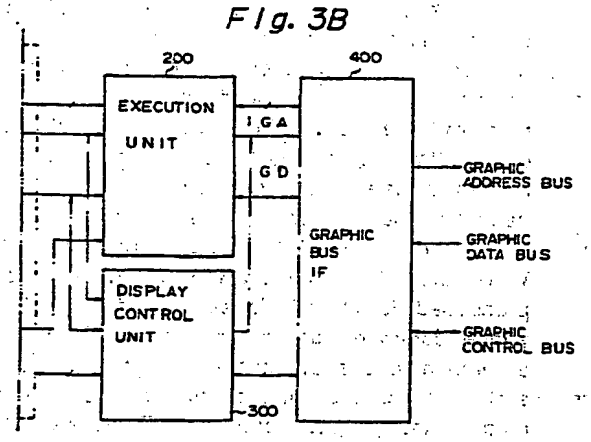


Fig. 3B



BNSDOCID: <EP___0389175A2_I_>

DATA PREFETCH SYSTEM

BACKGROUND OF THE INVENTION

(1) Field of the Invention

The present invention relates to a data processing system wherein data prefetch is carried out, and a data processing device which is used in the system, having a function for prefetching data, and including a buffer memory wherein prefetched data is temporarily stored. The data which is prefetched includes commands and parameters which accompany respective commands. The parameters include numerical information, and are respectively processed in accordance with the corresponding commands.

The present invention is particularly advantageous for a data processing device which is used in a data processing system, which is connected to a common bus, and which shares the use of the common bus with other bus masters such as a central processor. Further, the present invention is applicable to a co-processor which is provided to take a part of the work of a central processor to enhance the capability of the central processor, and a co-processor system which contains the above co-processor therein. Typical examples of such co-processors are an image processor and an arithmetic processor. In addition, the above co-processor may be constructed within a LSI chip as a peripheral LSI which cooperates with a microprocessor unit (MPU).

(2) Description of the Related Art

In data processing devices which process data, the data which is to be processed is supplied from a memory under control of a central processor, or is fetched from the memory by the data processing device itself, for example, by a DMA operation. In the data processing device having a function for prefetching data, the prefetched data is temporarily stored in a command and parameter buffer memory which is provided therein, in both cases the data is supplied under control of a central processor, and is prefetched by the data processing device itself.

In a data processing system wherein a common bus is shared by a plurality of bus masters, a bus arbitration is carried out for controlling use of the common bus by the plurality of bus masters, and it takes a considerable time to carry out the operation of the bus arbitration. During the operation of the bus arbitration, data transfer cannot be carried out using the bus. Therefore, when the frequency of the bus arbitration increases, the efficiency regarding the use of the bus, and thus the efficiency of the system is lowered.

Thus, when the above data processing device having a function for prefetching data, is incorporated in the above data processing system wherein a common bus is shared by a plurality of bus masters, reduction of the frequency of the prefetch operation is required to prevent the lowering of the efficiency of the system.

However, in the prior art, the above data prefetch in the data processing device is carried out every time a vacancy for storing a new data is generated in the command and parameter buffer memory. The vacancy in the command and parameter buffer memory is generated every time an amount of data is read out from the command and parameter buffer memory, and is processed in an executing portion of the data processing device. When the above memory wherein the data to be processed is stored, is connected through the above common bus, operations for obtaining an allowance to use the bus with regard to the bus arbiter, are carried out every time before the beginning of the prefetch operation. Therefore, in the conventional data processing device, the frequency of bus arbitration is increased by the above prefetch procedure.

In addition, in the prior art, the prefetch is carried out simply in the order of successive addresses in the memory wherein the data to be processed is stored, since, generally, the data to be processed is stored in the order of the processing. However, the data may include a branch command. When a branch command is processed in the data processing device, in most cases, the data which is to be processed in the next step, is not the data which is stored in the address next to the branch command in the above memory address, and therefore, is not the data which has been prefetched following the branch command and stored in the command and parameter buffer memory. Namely, when a branch command appears in the data which is to be processed, the prefetched data following the branch command is, in most cases, useless. This means that useless prefetch operations and the useless bus arbitrations are carried out in the conventional data processing device, and these useless bus arbitrations lower the efficiency of the system.

Further, generally, the above data includes commands each of which instructs what kind of processing is to be carried out, and one or more parameters which accompany the command and

are to be used for the processing. Accordingly, in the prior art, the data which is fetched in the data processing device is comprised of a succession of pairs of a command and accompanying parameters' set. However, often, the pairs including the same command and different parameters, are successively fetched. The fetched command is decoded at the first stage of a processing of each command in the data processing device even while data (the above pairs) including the same commands are successively processed. It is not desirable to repeat the prefetch and decoding of the same command regarding the above-mentioned bus efficiency and an efficiency of the data processing device itself.

SUMMARY OF THE INVENTION

A first object of the present invention is to provide a data processing system carrying out data prefetch, and a data processing device used in the system, wherein a frequency of the operations for obtaining an allowance to use a common bus, is reduced.

A second object of the present invention is to provide a data processing system carrying out data prefetch, and a data processing device used in the system, having a function for prefetching data including a command and parameter buffer memory for storing the prefetched data, wherein the speed and efficiency of data processing is improved.

A third object of the present invention is to provide a data processing system carrying out data prefetch, and a data processing device used in the system, which data processing device takes a part of the work of a central processor to enhance the capability of the central processor when initializing the registers therein.

A fourth object of the present invention is to provide a data processing device having a function for prefetching data, including a command and parameter buffer memory for storing the prefetched data, wherein information on data which is processed, is held when an exceptional condition arises.

According to the present invention, there is provided a data processing device contains an executing portion and a data buffer circuit. The above data buffer circuit stores one or more commands and/or one or more parameters which are prefetched, until each of the commands and parameters is read out by the above executing portion. The above executing portion inputs a command which is the oldest one of the commands stored in the data buffer circuit when an execution of a preceding command is completed, inputs one or more parameters stored in the data buffer circuit when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein. The above data processing device further contains an insufficient space detecting circuit for detecting a first state of the data buffer circuit in which state there is not enough vacant space in the data buffer circuit in which a further command and/or a parameter can be stored, and a least data detecting circuit for detecting a second state of the data buffer circuit in which state the data buffer circuit does not store data including a command and/or a parameter, which is necessary for a next operation in the above executing circuit. The system using the above device contains a memory which stores the above commands and parameters in the order of execution in the executing circuit. A prefetch control circuit is provided either in the device or outside of the device.

A first additional feature of the present invention is that the above prefetch control circuit comprises a branch command detecting circuit for detecting a branch command in prefetched commands, and the prefetch control circuit stops the prefetch operation until a result of the execution of the branch command is obtained in the execution circuit.

A second additional feature of the present invention is that, in the above data processing system containing the above data processing device, when successive executions of a command by the data processing device are required, which executions each use a parameter set comprised of one or more parameters, the command is stored in the command and parameter memory, followed by the parameter sets which are respectively used in successive executions of the command; and the command contains information on the number of the parameter sets, and information on the data length of each of the parameter sets.

A third additional feature of the present invention is that, in the above data processing device, the above prefetch control circuit comprises an address calculation circuit for calculating an address for prefetching a command or a parameter, and the address calculation circuit comprises a command address FIFO memory and an executing command address register. The above command address FIFO memory inputs the output of the address calculation circuit, and outputs its own oldest content into the executing command address register to hold the address of a command which is currently in execution, where the content of the executing command address register is renewed responding to an input of a new command into the execution circuit.

A fourth additional feature of the present invention is that, in the above data processing device,

BNSDOCID: <EP___0389175A2_I_>

the above prefetch control circuit comprises a command register for temporarily holding a prefetched command, and the command register comprises a command FIFO memory and an executing command register. The command FIFO memory inputs the output of the command register, and outputs its own oldest content into the executing command register to hold a command which is currently in execution, whereupon the content of the executing command register is renewed responding to an input of a new command into the execution circuit.

A fifth additional feature of the present invention is that, the above data processing device further comprises one or more internal registers which are requested to be initialized. One or more initialization commands can be stored in the command and parameter memory, where the initialization commands respectively contain internal addresses of the internal registers; and the initialization commands each can be followed by parameter data which contains initialization data which is to be written in a corresponding internal register, in the command and parameter memory. The prefetch control circuit comprises: a command decode circuit for determining whether or not a command fetched from the command and parameter memory is the initialization command; a parameter register for receiving the initialization data therein; and an address field register for holding the internal address contained in the initialization command; an address decoder circuit for decoding its own input to output an input control signal to the internal register which is addressed by the internal address; a switching circuit for outputting either of an address signal supplied from outside of the data processing device, and the output of the address field register; and an initialization control circuit for controlling the switching circuit to select the output of the address field register, as its own output, when an initialization command is determined.

A sixth additional feature of the present invention is that, in the above data processing device, the execution circuit has a first port, connected to a first bus through the prefetch control circuit, for inputting the commands and parameters, and a second port, connected to a second bus, for outputting results of the execution.

BRIEF DESCRIPTION OF THE DRAWINGS

In the drawings:

Figure 1 shows a portion of a data processing system having a function for processing and displaying image data, to which an embodiment of the present invention is applied;

Figure 2 shows timing diagrams of the above fetching operation of the commands and parameters, and the outputting operation of the processed data, in the conventional construction, and in the construction of Fig. 1;

Figures 3A and 3B show a construction of the image processor 500 of Fig. 1 according to an embodiment of the present invention;

Figure 4 shows an example of a conventional structure of a series of commands and parameters accompanying respective commands;

Figure 5 shows an example of the content of the conventional command and parameter memory corresponding to the above command and parameter structure of Fig. 4;

Figure 6 shows a timing of executing data processing which is instructed by the series of commands and parameters as shown in Fig. 5;

Figure 7 shows an example of the structure of a command and a series of parameters accompanying the command;

Figure 8 shows an example of the content of the command and parameter memory 502 corresponding to the above command and parameter structure of Fig. 7;

Figure 9 shows a timing of executing data processing which is instructed by the series of commands and parameters as shown in Fig. 7;

Figure 10 shows the format of the command word CW in the embodiment of the present invention;

Figure 11 shows the construction of the command decoding portion 50 in the image processor 500 of Fig. 3A;

Figure 12 shows the construction of the address calculation circuit 40 in the image processor 500 of Fig. 3A;

Figure 13 shows an example of the timing of the operation in the embodiment when a branch command is prefetched;

Figure 14 shows the construction of the parameter counting portion 60 in the image processor 500 of Fig. 3A;

Figures 15A and 15B, and 16 show the constructions of the execution unit interface 80 in the image processor 500, the command and parameter buffer memory for storing the prefetched command sets, and the data amount monitoring circuit provided in the execution unit 200;

Figures 17A, 17B, and 17C show the control operation by the micro ROM 71 in the embodiment of the present invention;

Figure 18 shows the control operation by the execution portion 224 in the embodiment of the present invention;

Figure 19 shows an example of timing of the operation in the construction of Fig. 16;

Figures 20 to 23 show examples of various modes of the structures of command sets, other than the structure shown in Fig. 7; and

BNSDOCID: <EP 0389175A2 >

Figure 24 shows an additional construction for initializing internal registers in the image processor 500.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

(1) Whole Data Processing System

Figure 1 shows a portion of a data processing system having a function for processing and displaying image data, to which an embodiment of the present invention is applied.

In Fig. 1, reference numeral 501 denotes a CPU, 502 denotes a command and parameter memory, 500 denotes an image processor, 503 denotes a graphic memory, 504 denotes a CRT display device, 505 denotes a system bus, and 506 denotes a graphic bus.

The image processor 500 is provided for carrying out processing of image data to display an image according to an instruction from the CPU 501, e.g., to draw a drawing on the display device, or the like.

Commands and parameters accompanying the respective commands, which are fetched and processed in the image processor 500 are stored in the command and parameter memory 502, basically in the order of the processing. The content of the command and parameter memory 502 is written by the CPU 501 in advance of the operation of the image processor 500.

The CPU 501 and the command, parameter memory 502, and the one side of the image processor 500 are connected to the system bus 505, and the above commands and parameters accompanying the respective commands, are fetched through the system bus 505 from the command and parameter memory 502 to the image processor 500.

The graphic memory 503, the CRT display device 504, and the other side of the the image processor 500 are connected to the graphic bus 506, and the processed data which is in the bitmap form is output from the the image processor 500 to the the graphic memory 503 through the graphic bus 506.

Although not shown, conventionally, all of the CPU 501, the command and parameter memory 502, the image processor 500, the graphic memory 503, and the CRT display device 504 are connected to a common bus, and therefore, the above fetching operation of the commands and parameters, and the outputting operation of the processed data cannot be carried out in parallel in the image processor 500 because both the operations use the same bus. However, in the construction as shown Fig. 1, wherein the graphic bus 506 for transferring the processed data, is provided separately from the system bus 505 for transferring the command and parameters, the above fetching operation of the commands and parameters, and the outputting operation of the processed data can be carried out in parallel in the image processor 500.

Figure 2 shows timing diagrams of the above fetching operation of the commands and parameters, and the outputting operation of the processed data, in the conventional construction, and in the construction of Fig. 1.

In Fig. 2, TO1 indicates a timing of the operation for fetching commands and parameters in the conventional construction, TO2 indicates a timing of the operation for outputting the processed data in the conventional construction, TN1 indicates a timing of the operation for fetching commands and parameters in the construction of Fig. 1, and TN2 indicates a timing of the operation for outputting the processed data in the construction of Fig. 1.

As indicated in Fig. 2, the fetching operation of the commands and parameters, and the outputting operation of the processed data are carried out in parallel in the construction of Fig. 1, and therefore, a total processing time is reduced.

(3) Whole Construction of Image Processor

Figures 3A and 3B show a construction of the image processor 500 of Fig. 1 according to an embodiment of the present invention.

In Figs. 3A and 3B, reference numeral 100 denotes a prefetch unit, 200 denotes an execution unit, 300 denotes a display control unit, and 400 denotes a graphic bus interface. In the construction of the prefetch unit 100, reference numeral 10 denotes an access control circuit, 20 denotes an address data input/output interface circuit, 30 denotes an interrupt control circuit, 40 denotes an address calculation circuit, 50 denotes a command decoding portion, 60 denotes a parameter counting portion, 70 denotes a command fetch control portion, 80 denotes an execution portion, 91 denotes an internal address bus, and 92 denotes an internal data bus.

The prefetch unit 100 carries out the prefetch operation of commands and parameters from the aforementioned command and parameter memory 502.

The execution unit 200 executes data processing operation using the fetched parameters in accordance with the fetched commands.

The display control unit 300 generates control signals for the CRT display device 504, e.g., horizontal and vertical synchronizing signals.

BNSDOCID: <EP___0389175A2_I_>

The graphic bus interface 400 is provided as an interface between the the graphic bus 506 and the above internal circuits of the image processor 500.

In the construction of the prefetch unit 100, command decoding portion 50 receives a prefetched command, and decodes the command. The construction and the operation of the parameter counting portion 60 are explained. The command fetch control portion 70 comprises a microprogram ROM, and controls the whole operation of the prefetch unit 100. The execution unit interface 80 generates control signals to control the execution unit 200 as explained later.

The address calculation circuit 40 calculates addresses for prefetching commands and parameters from the command and parameter memory 502, under the control of the command fetch control portion 70.

The interrupt control circuit 30 receives interrupts, for example, from the CPU 501 for reading the content of internal registers in the image processor 500, or from the the execution unit 200 to inform of an occurrence of an exceptional condition. These interrupts are transmitted to the the command fetch control portion 70, and the command fetch control portion 70 outputs control signals corresponding to the respective interrupts.

The address data input/output interface circuit 20 is provided as an interface between the system bus 505 and the internal address and data buses 91 and 92. The address calculated in the the address calculation circuit 40 is first output in the internal data bus 92, and the address signal on the internal data bus 92 is switched onto an address bus in the system bus 505 to apply the address signal.

The access control circuit 10 comprises a DMA controller, and controls a DMA transfer operation for prefetching commands and parameters from the command and parameter memory 502, under the control of the command fetch control portion 70. For example, the access control circuit 10 outputs a request for an allowance to use the system bus 505 to the bus arbiter, receives an allowance to use the system bus 505 from the bus arbiter, controls the outputting of the address signal to the command and parameter memory 502, and write timing signal WR to receive and hold a prefetched word, which is prefetched from the command and parameter memory 502, in the image processor 500.

(4) Conventional Structure of Commands and Parameters

Figure 4 shows an example of conventional structure of a series of commands and parameters accompanying respective commands.

In Fig. 4, C1, C2, ... Cn each denote a command set, OP denotes an operation word, and P_{11}, ... P_{1m}, P_{21}, ... P_{2m}, ... P_{n1}, ... P_{nm} each denote a parameter word, where "n" is the number of the command sets, and "m" is the number of the parameter words in each command set. Each command set comprises a command word, and zero to a plurality of parameter words, and each command word contains an operation code which indicates what the command instructs. A set of parameters which is contained in the in m parameter words P_{i1}, ... P_{im} (i = 1 -m) following each command word Ci, are used at the same time for executing one operation in the execution unit 200 in accordance with the command word Ci.

Figure 5 shows an example of the content of the conventional command and parameter memory corresponding to the above command and parameter structure of Fig. 4.

In the example of Fig. 5, a series of command sets for line segment generating command is shown. In the address 0000, a command word which indicates that the command is a line segment generating command, is stored. In the address 0001, the X and Y coordinates of starting point (X_{S0}, Y_{S0}) are stored, and in the address 0002, the X and Y coordinates of starting point (X_{E0}, Y_{E0}) are stored. In the following addresses 0011, 0100, ..., similar sets of commands and accompanying parameters to draw line segments in different coordinates are stored.

Figure 6 shows a timing of executing data processing which is instructed by the series of commands and parameters as shown in Fig. 5.

As shown in Fig. 6, in the conventional data processing system wherein commands and parameters supplied to the image processor have a structure as shown in Fig. 5, a command decoding operation is necessary for a timing before executing each data processing operation of a set of the coordinates of the starting points and the ending points of a line segment in the execution unit 200 even when the same command (the line segment generating command) is successively executed, since the command word is fetched for each set of the coordinates of the starting points and the ending points of a line segment. Namely, in the conventional data processing system wherein commands and parameters having a structure as shown in Fig. 4, a command decoding operation is necessary for a timing before executing each data processing operation using a set of parameters in the execution unit 200 even when the same command is successively executed.

(5) Structure of Commands and Parameters in the

RNSDOCID: <EP 0389175A2 I >

Embodiment of Present Invention

Figure 7 shows an example of the structure of a command and a series of parameters accompanying the command. The command and the parameters in Fig. 7 correspond to the the series of the commands and the parameters in Fig. 4, i.e., the command and the parameters in Fig. 7 have the same information as the the series of the commands and the parameters in Fig. 4 when the contents in the command words C1, C2, ... Cn are the same in Fig. 4.

In Fig. 7, CW denotes a command word, BP1, BP2, ... BPn each denote a basic parameter set, OP denotes an operation code, and P_{11}, ... P_{1m}, P_{21}, ... P_{2m}, ... P_{n1}, ... P_{nm} each denote a parameter word as in Fig. 4, where "n" is the number of the basic parameter sets, and "m" is the number of the parameter words in each basic parameter set. Generally, each command word CW is accompanied by zero to a plurality of basic parameter sets, and each command word CW contains an operation code OP which indicates what the command instructs, the number BP of the parameter words in each basic parameter set, and the number NP of the basic parameter sets. In the example of Fig. 7, BP = m and NP = n. A set of parameters which is contained in the in m parameter words P_{i1}, ... P_{im} (i = 1 - m) in each basic parameter set BPi, are used at the same time for executing one operation in the execution unit 200 in accordance with the command word OP.

Figure 8 shows an example of the content of the command and parameter memory 502 corresponding to the above command and parameter structure of Fig. 7.

In the example of Fig. 8, a command set of a line segment generating command is shown. In the address 0000, a command word which indicates that the command is a line segment generating command, is stored. In the address 0001, the X and Y coordinates of starting point (X_{S0}, Y_{S0}) are stored, and in the address 0002, the X and Y coordinates of starting point (X_{E0}, Y_{E0}) are stored. In the following pairs of addresses (0011, 0100), (0101, 0111), ... , similar sets of parameters each pair containing a pair of the coordinates of a starting point and an ending point, are stored to draw line segments in different coordinates.

Figure 9 shows a timing of executing data processing which is instructed by the series of commands and parameters as shown in Fig. 7.

As shown in Fig. 9, in the data processing system wherein a command and parameters supplied to the image processor 500 have a structure as shown in Fig. 8, a command decoding operation is necessary once just after the command word CW is input in the execution unit 200, i.e., the command decoding operation is not necessary for a timing before executing each data processing operation of a set of the coordinates of the starting points and the ending points of a line segment in the execution unit 200 when the same operation (the line segment generating command) is repeatedly executed for different sets of parameters, since the command word is fetched at the top of the command set only.

Thus, when the same operation is repeatedly executed for different sets of parameters, the time necessary to fetch the total set of commands and parameters, and to decode the command are reduced by using the set of a command and parameters as shown in Fig. 7, and further, it reduces the necessary size of the command and parameter memory 502, and the time occupying the system bus 505 for fetching the above set.

Figure 10 shows the format of the command word CW in the embodiment of the present invention.

In Fig. 10, OP CODE denotes an operation code, BP denotes the above-mentioned number of the parameter words in each basic parameter set, and NP denotes the above-mentioned number of the basic parameter sets. The denotations MODE and REGISTERS ADDRESS are explained later.

(6) The Command Decoding Portion 50

Figure 11 shows the construction of the command decoding portion 50 in the image processor 500 of Fig. 3A.

In Fig. 11, reference numeral 51 denotes a command register, 52 denotes a parameter register, 53 denotes a command decoder, 55 denotes a FIFO memory, and 56 denotes an executing command register. The micro ROM 71 shown in Fig. 11 is a main component of the command fetch control portion 70.

The command register 51 and the parameter register 52 are connected to the internal data bus 92. The above-mentioned command word CW shown in Fig. 10 is written in the command register 51 when the command word CW is prefetched. The output of the command register 51 is decoded in the command decoder 53. Among the output of the command decoder 53, the decoded result of the operation code OP CODE is applied to the input terminal of the micro ROM 71, and the decoded results of the numbers BP and NP, and the portion MODE are input into the parameter counting portion 60. The content of the potion REGISTER ADDRESS in the command word CW is applied to an address field register 521 which is explained later with reference to Fig. 24.

The output of the command register 51 is also

BNSDOCID: <EP___0389175A2_I_>

input into the FIFO memory 55. The FIFO memory 55 and the executing command register 56 are provided to hold the command code when an exceptional condition occurs. As explained later, all the commands and the parameters (command sets) which have been prefetched are stored in the command and parameter buffer memory in the execution unit 200 until the data processing regarding the command or the parameters are executed in the the execution unit 200. The command codes stored in the command and parameter buffer memory in the execution unit 200 are at the same time held in the FIFO memory 55, and the command word regarding which the data processing is executed in the execution unit 200, is held in the executing command register 56. To realize the above, a signal CEXE which is output from the execution unit 200 when the execution unit 200 inputs a new command to be executed, is applied to the FIFO memory 55 and the executing command register 56, for an output control of the FIFO memory 55 and for an input control of the executing command register 56, respectively. Regarding the exception processing, a further explanation is given later.

(7) The Address Calculation Circuit 40

Figure 12 shows the construction of the address calculation circuit 40 in the image processor 500 of Fig. 3A.

In Fig. 12, reference numerals 41 and 43 each denote a selector, 42 denotes an adder circuit, 44 denotes a prefetch address pointer, 45 denotes a FIFO memory, and 46 denotes an executing command address register.

The selector 41 receives the value from the internal data bus 92, a constant value +1, and a constant value -1, and selects one of the inputs as its output under the control of the micro ROM 71. The output of the selector 41 is applied to one of the input terminals of the adder 42. The prefetch address pointer 44 holds an address which is used for a prefetch operation. The output of the prefetch address pointer 44 is output to the internal data bus 92, and is applied to the other input terminal of the adder 42. The adder 42 adds the above two inputs. The selector 43 receives the value from the internal data bus 92 and the output of the adder 42, and one of the inputs as its output under the control of the micro ROM 71. The output of the selector 42 is applied to the prefetch address pointer 44.

Normally, the selector 41 selects +1, and the selector 43 selects the output of the adder 42 as their outputs, respectively. When a command which is decoded in the command decoder 53 is a branch command which instructs to jump the address by a relative address value, the micro ROM 71 controls the selector 41 to select the input from the internal data bus 92, and the selector 43 to select the output of the adder 42. When a command which is decoded in the command decoder 53 is a branch command which instructs to jump the address to an absolute address, the micro ROM 71 controls the selector 43 to select the input from the internal data bus 92.

The (relative or absolute, or direct or indirect) address information contained in the branch command is once written in the command register 51 and/or the parameter register 52 when the branch command is prefetched. Although not shown, branch commands are constituted by a command word only, or a command word and a parameter word in this embodiment. When a branch command is constituted by a command word only, the command word is written in the command register 51, or when a branch command is constituted by a command word and a parameter word, the command word is written in the command register 51, and the parameter word is written in the parameter register 52. When information on direct addressing is contained in the command, the address information in the command is supplied to the address calculation circuit 40 through the internal data bus 92, or when information on indirect addressing is contained in the command, the relative or absolute address value is obtained by fetching the value from the memory or a register using the address information in the command, and the obtained address value is supplied through the internal data bus 92 to the address calculation circuit 40. Further, the characteristic feature of the present invention regarding the operation when a branch command is prefetched, is explained later.

The output of the prefetch address pointer 44 is also input into the FIFO memory 45. The FIFO memory 45 and the executing command register 46 are provided to hold the address of the command code when an exceptional condition occurs. As mentioned before, all the commands and the parameters (command sets) which have been prefetched are stored in the command and parameter buffer memory in the execution unit 200 until the data processing regarding the command or the parameters are executed in the the execution unit 200. The address of the command codes which codes are stored in the command and parameter buffer memory in the execution unit 200 are at the same time held in the FIFO memory 45, and the address of the command word regarding which the data processing is executed in the execution unit 200, is held in the executing command register 46. To realize the above, a signal CEXE which is output from the execution unit 200 when the execu-

BNSDOCID: <EP 0389175A2 I >

tion unit 200 inputs a new command to be executed, is applied to the FIFO memory 45 and the executing command register 46, for an output control of the FIFO memory 45 and for an input control of the executing command register 46, respectively. Regarding the exception processing, a further explanation is given later.

(8) The Operation When a Branch Command is Prefetched

As explained before, in the prior art, the prefetch is carried out simply in the order of successive addresses in the memory (the command and parameter memory 502) wherein the data (commands and parameters) to be processed is stored, since, generally, the data to be processed is stored in the order of the processing (as shown in Figs. 5 and 8). In the prior art, when the branch command is processed in the data processing device, in most cases, the command which is to be processed in the next step, is not the command which is stored in the address following the branch command in the above memory, and therefore, is not the command which has been prefetched following the branch command and is stored in the command and parameter buffer memory. Namely, when a branch command is detected as an output of the command decoder 53, the prefetched data following the branch command is, in most cases, useless. This means that useless prefetch operations and the useless bus arbitrations are carried out in the conventional data processing device, and these useless bus arbitrations lower the efficiency of the system.

The branch commands includes: simple jump commands (containing information on an absolute address or a relative address); conditional jump commands; subroutine jumps (containing information on an absolute address or a relative address); and return subroutine jump commands, and the like. Generally, where to jump by a branch command is unknown until the execution of the branch command is completed in the execution unit 200.

According to the present invention, the micro ROM 71 temporarily stops the prefetch operation for the commands following the branch command when a branch command is detected as an output of the command decoder 53. The temporary stop continues until the execution of the branch command is completed in the execution unit 200.

An example of the timing of the operation in the embodiment when a branch command is prefetched, is shown in Figure 13. At the time t_7, a jump command word JMP is prefetched and decoded. From the decoded result, it is known that a parameter word P31, which contains the aforementioned address information, follows the jump command word JMP, in this example. Accordingly, the micro ROM 71 controls prefetching of the next one word which is stored in the address next to the address wherein the jump command word JMP is stored in the command and parameter memory 502. Then, at the time t_8, the micro ROM 71 stops the prefetch operation for the commands following the branch command until the execution of the branch command is completed in the execution unit 200. At the time t_{11}, the execution of the branch command is completed in the execution unit 200. As the result of the execution of the branch command, information on the address for the next prefetch operation is given to the address calculation circuit 40 through the internal data bus 92, and the micro ROM 71 starts the following prefetch operation.

Thus, according to the present invention, useless prefetch operations and the useless bus arbitrations are minimized, and therefore, the efficiency of the system is improved.

(9) The Parameter Counting Portion 60

Figure 14 shows the construction of the parameter counting portion 60 in the image processor 500 of Fig. 3A.

In Fig. 14, reference numeral 61 denotes a gate circuit, 62 denotes a basic parameter counter, 63 denotes a selector, 64 denotes a repeating number counter, 65 denotes a repeating number register, 66 denotes a shifter, 67 denotes an OR circuit, 68 denotes a executing parameter number register, 69 denotes a FIFO memory, and 79 denotes an accord detecting circuit. CNT denotes a count signal which is output every time when one parameter word is prefetched from the command and parameter memory 502, and is generated by the micro ROM 71.

The gate 61 is provided in the way of the count signal CNT to the basic parameter counter 62 as a count down signal. The selector 63 receives the count signal CNT and the output of the basic parameter counter 62, and selects one of these inputs as its output to the repeated number counter 64 as a count up signal. The shifter 66 is provided in the way of the aforementioned number NP of the basic parameter sets in a command set, which is supplied from the aforementioned command decoder 53.

When a new command word is prefetched and decoded in the command decoding portion 50, the aforementioned number BP of the parameter words in each basic parameter set, which is supplied from the aforementioned command decoder 53, is loaded in the basic parameter counter 62 as a preset

BNSDOCID: <EP___0389175A2_I_>

value, and the above number NP is loaded in the repeated number register 65 through the shifter 66 as a preset value. The gate circuit 61, the selector 63, and the shifter 66 are controlled by the aforementioned portion MODE in the content of the command word as shown in Fig. 10.

When the command set has a structure as shown in Fig. 7, the gate 61 is open, the selector 63 selects the output of the basic parameter counter 62 as its output, and the shifter does not shift the above number NP.

The count of the basic parameter counter 62 indicates the number of parameter words which have not been prefetched yet in each basic parameter set. When all the parameter words in each basic parameter set have been prefetched, the basic parameter counter 62 outputs an active parameter zero signal. The parameter zero signal also become active when a command word which is not followed by a parameter word, is prefetched, as the BP portion in the command word indicates zero.

The count of the repeated number counter 64 indicates the number of basic parameter sets which have been already prefetched in each command set. The accord detecting circuit 79 compares the count of the repeated number counter 64 and the output of the repeated number register 65. When all the basic parameter sets in each command set have been prefetched, the accord detecting circuit 79 outputs a command set prefetch complete signal NPEND. The logical sum of the output of the basic parameter counter 62, and the output of the accord detecting circuit 79 is obtained in the OR circuit 67, and is output as a signal BPEND to the micro ROM 71 in the command fetch control portion 70 and to the execution unit interface 80. Thus, the BPEND signal becomes active every time a basic parameter set is prefetched (a prefetch of a basic parameter set is completed), or when a command word which is not followed by a parameter word, is prefetched.

The output of the repeated number counter 64 is also input into the FIFO memory 69. The FIFO memory 69 and the executing parameter number register 68 are provided to hold the number of basic parameter sets which have been already prefetched in each command set when an exceptional condition occurs. As mentioned before, all the commands and the parameters (command sets) which have been prefetched are stored in the command and parameter buffer memory in the execution unit 200 until the data processing regarding the command or the parameters are executed in the the execution unit 200. The number of the basic parameter sets which are stored in the command and parameter buffer memory in the execution unit 200 are at the same time held in the FIFO memory 69, and the number of the basic parameter set regarding which the data processing is executed in the execution unit 200, is held in the executing parameter number register 68. To realize the above, a signal PEXE which is output from the execution unit 200 when the execution unit 200 inputs a new basic parameter set to be executed, is applied to the FIFO memory 69 and the executing parameter number register 68, for an output control of the FIFO memory 69 and for an input control of the executing command register 68, respectively. Regarding the exception processing, a further explanation is given later.

Further, the operation of the parameter counting portion 60 when a command set has a different structure from the structure shown in Fig. 7, will be explained later.

(10) The Execution Unit Interface 80 and The Data Buffer in the Execution Unit 200.

Figures 15A and 15B show the constructions of the execution unit interface 80 in the image processor 500, the command and parameter buffer memory for storing the prefetched command sets, and the data amount monitoring circuit provided in the execution unit 200.

In Figs. 15A and 15B, the construction encircled by a dashed line 80 is the execution unit interface 80 of Fig. 3A, and the construction encircled by a dashed line 210 is the command and parameter buffer memory for storing the prefetched command sets. Reference numeral 220 denotes an executing command set holding portion wherein a command set regarding which the execution unit 200 is carrying out data processing, is held, 230 denotes a data amount monitoring circuit which cooperates with the execution unit interface 80, and 224 denotes an execution portion which executes data processing using the basic parameter set held in the executing command set holding portion 220 in accordance with a command held in the executing command set holding portion 220.

The command and parameter buffer memory 210 comprises a command FIFO memory 211 for storing prefetched command words, and a parameter FIFO memory 212 for storing prefetched basic parameter sets.

The executing command set holding portion 220 comprises an executing command register 221 for holding a command word in execution, an executing command decoder 222 for decoding the content of the executing command register 221, and an executing basic parameter set register 223 for holding a basic parameter set in execution.

The data amount monitoring circuit 230 monitors the amount of the prefetched command words stored in the command FIFO memory 211, and the

EP 0 389 175 A2

amount of the prefetched basic parameter sets stored in the parameter FIFO memory 212, and outputs a prefetch allowance signal WRRDY, which is supplied to the micro ROM 71 in the command fetch control portion 70.

The execution unit interface 80 generates write control signals for the command and parameter buffer memory 210, and outputs information on the prefetch of the command words and parameter words, respectively, to the data amount monitoring circuit 230.

The execution unit interface 80 comprises AND circuits 81, 83, 85. 86, and 87, an inverter 82, and a C/P flag register 84.

The C/P flag register 84 indicates whether a word prefetched at the time is a command word or a parameter word, and is realized by a flip-flop circuit. The C/P flag register 84 receives a CSET signal and a reset signal RESET from the micro ROM 71 at its reset terminal, and a PSET signal from the micro ROM 71 at its set terminal. The $\overline{Q}$ output C/P of the C/P flag register 84, which is "1" when the C/P flag register 84 is in the set state, or "0" when the C/P flag register 84 is in the reset state, is applied directly to one input terminal of the AND circuit 81, is applied through the inverter 82 to one input terminal of the AND circuit 83, and is further applied to the micro ROM 71 in the command fetch control portion 70. The aforementioned write timing signal WR, which is output from the access control circuit 10 for hold a prefetched word in the image processor 500, is applied to the other input terminals of both the AND circuits 81 and 83. Thus, the AND circuit 81 outputs an active signal when a command word is prefetched, and the AND circuit 83 outputs an active signal when a parameter word is prefetched.

The output of the AND circuit 81 is applied to the command register 51 as an input (write) control signal, and the output of the AND circuit 83 is applied to the parameter register 52 as an input (write) control signal.

The output of the AND circuit 81 is also applied to one input terminal of the AND circuit 85, and the output of the AND circuit 83 is also applied to one input terminal of the AND circuit 86. A part of the output of the command decoder 53 in the command decoding portion 50, which indicates whether or not the prefetched command code is a command which is to be executed in the execution unit 200, is applied to the other input terminal of each of the AND circuits 85 and 86. Thus, the AND circuit 85 outputs an active signal CMWR when a command word which is to be executed in the execution unit 200, is prefetched, and the AND circuit 86 outputs an active signal PRWR when a parameter word which is to be executed in the execution unit 200, is prefetched.

The above output CMWR of the AND circuit 85 is applied to the command FIFO memory 211 as an input control signal, and the above output PRWR of the AND circuit 86 is applied to the parameter FIFO memory 212 as an input control signal. The command FIFO memory 211 inputs a command word which is transferred from the command and parameter memory 502 through the internal data bus 92 when an active CMWR signal is applied thereto, and the parameter FIFO memory 212 inputs a parameter word which is transferred from the command and parameter memory 502 through the internal data bus 92 when an active PRWR signal is applied thereto.

The above part of the output of the command decoder 53 in the command decoding portion 50, which indicates whether or not the prefetched command code is a command which is to be executed in the execution unit 200, is also applied to one input terminal of the AND circuit 87. The aforementioned BPEND signal from the parameter counting portion 60 is applied to the other input terminal of the AND circuit 87. As the BPEND signal becomes active every time a basic parameter set is prefetched (a prefetch of a basic parameter set is completed), or when a command word which is not followed by a parameter word, is prefetched. Thus, the AND circuit 87 outputs an active signal every time a basic parameter set which is to be executed in the execution unit 200, is prefetched (a prefetch of a basic parameter set is completed), or when a command word which is to be executed in the execution unit 200, and is not followed by a parameter word, is prefetched.

The above output of the AND circuit 87 is supplied to the data amount monitoring circuit 230. The data amount monitoring circuit 230 comprises up-down counters 231, 233, and 235, a parameter vacancy detecting circuit 232, a command vacancy detecting circuit 234, a last operation amount detecting circuit 236, an OR circuit 237, and an RS-flip-flop circuit 238.

The up-down counter 231 receives the above PRWR signal from the AND circuit 86 in its UP input terminal, and a PEXE signal from the execution portion 224 in its DOWN input terminal. The execution portion 224 controls the parameter FIFO memory 212 and the executing basic parameter set register 223 to fetch a new basic parameter set from the parameter FIG memory 212 into the executing basic parameter set register 223, and outputs an active PEXE signal every time the fetch of a parameter word from the parameter FIFO memory 212 into the executing basic parameter set register 223 is carried out. Thus, the count PVC of the up-down counter 231 indicates the number of the parameter words which are stored in the parameter FIFO memory 212.

BNSDOCID: <EP___0389175A2_I_>

Similarly, the up-down counter 233 receives the above CWWR signal from the AND circuit 85 in its UP input terminal, and a CEXE signal from the execution portion 224 in its DOWN input terminal. The execution portion 224 controls the command FIFO memory 211 and the executing command register 221 to fetch a new command from the command FIFO memory 211 into the executing command register 221, and outputs an active CEXE signal every time the fetch of a command word from the command FIFO memory 211 into the executing command register 221 is carried out. Thus, the count CVC of the up-down counter 233 indicates the number of the command words which are stored in the command FIFO memory 211.

Further, the up-down counter 235 receives the output of the AND circuit 87 in its UP input terminal and an EAEXE signal from the execution portion 224 in its DOWN input terminal. The execution portion 224 outputs an active EAEXE signal every time a basic parameter set is fetched from the command and parameter buffer memory 210 into the executing command set holding portion 220 (a fetch of a basic parameter set is completed), or when a command word which is not followed by a parameter word, is fetched from the command and parameter buffer memory 210 into the executing command set holding portion 220. The fetch of a basic parameter set from the command and parameter buffer memory 210 into the executing command set holding portion 220, is carried out for each operation of data processing in the execution portion 224. Thus, the count EAC of the up-down counter 235 indicates the amount of the command data which are stored in the command and parameter buffer memory 210, by the number of operations which is necessary to complete the execution of the content of the command and parameter buffer memory 210.

The above count of the up-down counter 231 is input into the parameter vacancy detecting circuit 232, and the parameter vacancy detecting circuit 232 outputs an active signal when the count is more than a first predetermined number. The first predetermined number is determined as the following difference, (the maximum number of parameter words which can be stored in the the parameter FIFO memory 212) - (the possible maximum number of parameter words of a basic parameter set).

Similarly, the above count of the up-down counter 233 is input into the command vacancy detecting circuit 234, and the command vacancy detecting circuit 234 outputs an active signal when the count is more than a second predetermined number. The second predetermined number is determined as the following difference, (the maximum number of commands which can be stored in the the command FIFO memory 211) - (one).

The output of the parameter vacancy detecting circuit 232 and the output of the command vacancy detecting circuit 234 are input into the OR circuit 237. The output VC of the OR circuit 237 is applied to a reset terminal R of the RS-flip-flop circuits 238.

The above count of the up-down counter 235 is input into the last operation amount detecting circuit 236, and the last operation amount detecting circuit 236 outputs an active signal when the count is less than a third predetermined number. The third predetermined number is, for example, one or two.

The output of the last execution amount detecting circuit 236 is applied to a set terminal S of the RS-flip-flop circuit 238. The Q output of the RS-flip-flop circuit 238 is the aforementioned prefetch allowance signal WRRDY, which is supplied to the micro ROM 71. Thus, the prefetch allowance signal WRRDY is active ("1") when the number of vacant word spaces in the parameter FIFO memory 212 is larger than the possible maximum number of parameter words of a basic parameter set, and at least one vacant word space exists in the command FIFO memory 211, and an amount of data (command and/or parameter) corresponding to at least the above third predetermined number is stored in the command and parameter buffer memory 210.

Receiving the prefetch allowance signal WRRDY, the micro ROM 71 controls the prefetch unit to start a series of prefetch operations when the prefetch allowance signal WRRDY changes "1" to "0", and continue the prefetch operation until the prefetch allowance signal WRRDY returns to "0".

As a variation of the construction of Figs. 15A and 15B, the command FIFO memory 211 may be replaced by a register, and the parameter FIFO memory 212 may be replaced by a shift register having stages of a number equal to the number of the parameter words in a basic parameter set wherein only one command word and only one basic parameter set are held in the command and parameter buffer memory 210. In the above variation, the up-down counters 231 and 233 are each replaced by a flag register which indicates whether or not a corresponding register in the command and parameter buffer memory 210 is vacant, and the up-down counter 235 is unnecessary.

(11) Other Constructions of The Execution Unit Interface 80 and The Command and Parameter Buffer Memory in the Execution Unit 200

Figure 16 shows other constructions of the execution unit interface 80 in the image processor 500, the command and parameter buffer memory for storing the prefetched command sets, and the

data amount monitoring circuit provided in the execution unit 200.

In Fig. 16, the construction encircled by a dashed line 80′ corresponds to the execution unit interface 80 of Fig. 3A, and the construction encircled by a dashed line 213 corresponds to the command and parameter buffer memory for storing the prefetched command sets. Reference numeral 220′ denotes an executing command set holding portion wherein a command set regarding which the execution unit 200 is carrying out data processing, is held, 230′ denotes a data amount monitoring circuit which cooperates with the execution unit interface 80, and 224′ denotes an execution portion which executes data processing using the basic parameter set held in the executing command set holding portion 220 in accordance with a command held in the executing command set holding portion 220.

The command and parameter buffer memory 213 is constituted by a FIFO memory 211 for storing prefetched command words and prefetched basic parameter sets.

The executing command set holding portion 220′ comprises an executing command register 221 for holding a command word in execution, an executing command decoder 222 for decoding the content of the executing command register 221, an executing basic parameter set register 223 for holding a basic parameter set in execution, and a selector 225.

The data amount monitoring circuit 230′ monitors the amount of the prefetched data (command words and parameter words) stored in the FIFO memory 213, and outputs a prefetch allowance signal WRRDY, which is supplied to the micro ROM 71 in the command fetch control portion 70.

The execution unit interface 80′ generates a write control signal for the command and parameter buffer memory 213, and outputs information on the prefetch of the command or parameter words to the data amount monitoring circuit 230.

The execution unit interface 80 comprises AND circuits 85′ and 87′.

The aforementioned write timing signal WR, which is output from the access control circuit 10 for holding a prefetched word in the image processor 500, is applied to one input terminal of the AND circuit 85′. A part of the output of the command decoder 53 in the command decoding portion 50, which indicates whether or not the prefetched command code is a command which is to be executed in the execution unit 200, is applied to the other input terminal of each of the AND circuits 85′. Thus, the AND circuit 85′ outputs an active signal when a command word or a parameter word which is to be used for an execution of data processing in the execution unit 200, is prefetched.

The above output of the AND circuit 85′ is applied to the FIFO memory 213 as an input control signal, and the above output PRWR of the AND circuit 86 is applied to the parameter FIFO memory 212 as an input control signal. The FIFO memory 213 inputs a command or parameter word which is transferred from the command and parameter memory 502 through the internal data bus 92 when an active input control signal is applied thereto.

The AND circuit 87′ functions the same as the AND circuit 87 in Fig. 15A. Namely, the AND circuit 87′ outputs an active signal every time a basic parameter set which is to be executed in the execution unit 200, is prefetched (a prefetch of a basic parameter set is completed), or when a command word which is to be executed in the execution unit 200, and is not followed by a parameter word, is prefetched.

The above output of the AND circuit 87′ is supplied to the data amount monitoring circuit 230. The data amount monitoring circuit 230′ comprises up-down counters 235 and 239, a buffer vacancy detecting circuit 240, a last operation amount detecting circuit 236, and an RS-flip-flop circuit 238.

The up-down counter 239 receives the above output of the AND circuit 85′ in its UP input terminal, and a CPEXE signal from the execution portion 224′ in its DOWN input terminal. At the beginning of an execution of data processing for a command set, or when an execution of data processing for a command set is completed, the execution portion 224′ controls the FIFO memory 213, the selector 225, and the executing command register 221 to fetch a new command word from the parameter FIFO memory 212 into the executing command register 221. Or after a command word is fetched into the executing command register 221 and it is known that one or more parameter words follow the command word from the output of the command decoder 222, the execution portion 224′ controls the FIFO memory 213, the selector 225, and the executing basic parameter set register 223 to fetch a new basic parameter set from the FIFO memory 212 into the executing basic parameter set register 223. The execution portion 224′ outputs an active CPEXE signal every time the fetch of a word from the FIFO memory 213 into the executing basic parameter set register 223 is carried out. Thus, the count CPVC of the up-down counter 239 indicates the number of the command or parameter words which are stored in the FIFO memory 213.

Further, the up-down counter 235 receives the output of the AND circuit 87′ in its UP input terminal, and an EAEXE signal from the execution portion 224′ in its DOWN input terminal. The execution portion 224′ outputs an active EAEXE signal every time a basic parameter set is fetched from the command and parameter buffer memory 213

BNSDOCID: <EP___0389175A2_I_>

into the executing command set holding portion 220' (a fetch of a basic parameter set is completed), or when a command word which is not followed by a parameter word, is fetched from the command and parameter buffer memory 213 into the executing command set holding portion 220'. The fetch of a basic parameter set from the command and parameter buffer memory 213 into the executing command set holding portion 220', is carried out for each operation of data processing in the execution portion 224'. Thus, the count EAC of the up-down counter 235 indicates the amount of the command and parameter words which are stored in the command and parameter buffer memory 213, by the number of operations which is necessary to complete the execution of the content of the command and parameter buffer memory 213.

The above count CPVC of the up-down counter 239 is input into the buffer vacancy detecting circuit 240, and the buffer vacancy detecting circuit 240 outputs an active signal when the count is more than a predetermined number. The predetermined number is determined as the following difference, (the maximum number of command and parameter words which can be stored in the the FIFO memory 213) - (the possible maximum number of parameter words of a basic parameter set).

The output of the buffer vacancy detecting circuit 240 is applied to a reset terminal R of the RS-flip-flop circuit 238.

The above count of the up-down counter 235 is input into the last operation amount detecting circuit 236, and the last operation amount detecting circuit 236 outputs an active signal when the count is less than a third predetermined number. The third predetermined number is, for example, one or two.

The output of the last execution amount detecting circuit 236 is applied to a set terminal S of the RS-flip-flop circuit 238. The Q output of the RS-flip-flop circuit 238 is the aforementioned prefetch allowance signal WRRDY, which is supplied to the micro ROM 71. Thus, the prefetch allowance signal WRRDY is active ("1") when the number of vacant word spaces in the FIFO memory 213 is larger than the possible maximum number of words of a basic parameter set, and an amount of data (command and/or parameter) corresponding to at least the above fifth predetermined number is stored in the command and parameter buffer memory 213.

Receiving the prefetch allowance signal WRRDY, the micro ROM 71 controls the prefetch unit to start a series of prefetch operations when the prefetch allowance signal WRRDY changes "1" to "0", and continues the prefetch operation until the prefetch allowance signal WRRDY returns to "0".

(12) Control Operation in the Embodiment I

Figures 17A, 17B, and 17C show the control operation by the micro ROM 71 in the embodiment of the present invention. The operation of Figs. 17A, 17B, and 17C is written based on the construction of the execution unit interface 80 shown in Fig. 15A, but the operation of the construction of Figs. 17A, 17B, and 17C is applicable to the construction of Fig. 16 when some portions are changed as explained below.

Initially, in the step 600, the C/P flag register 84 is reset so that its $\overline{Q}$ output C/P becomes "1" by the micro ROM 71.

In the step 601, it is determined whether or not the prefetch allowance signal WRRDY is ON ("1"). When YES is determined, a prefetch operation of a next (command) word in the command and parameter memory 502, is commenced. In the step 603, when the command word is output (transferred) on the internal data bus 92, the access control circuit 10 outputs the aforementioned write control signal WR. Accordingly, the transferred command word is written in the command register 51 in the step 604. In the step 605, it is determined whether or not the command is a command which is to be executed in the execution unit 200. When it is determined YES in the step 605, the command word is written in the command and parameter buffer memory 210 in the execution unit 200, and the count CVC of the up-down counter 233 is incremented by one by the output CWWR of the AND circuit 85, in the step 606. When the command word is not followed by a parameter word, the count EAC of the up-down counter 235 is further incremented by one by the output of the AND circuit 87. When it is determined NO in the step 605, the step 606 is skipped. Then, in the step 607, the number BP of the parameter words in each basic parameter set, and the number NP of the basic parameter sets, which are contained in the command word and is supplied through the command decoder 53, are loaded as their preset count values in the basic parameter counter 62 and the repeated number counter 64, respectively.

In the step 609, it is determined whether or not the the output BPEND of the parameter counting portion 60 is "1". When it is determined NO, the C/P flag register 84 is set to make the $\overline{Q}$ output C/P "0" by the micro ROM 71, in the step 610. In the step 611, a prefetch operation of a parameter word following the previously prefetched word in the command and parameter memory 502, is commenced. In the step 612, the parameter word is output (transferred) on the internal data bus 92.

BNSDOCID: <EP 0389175A2 I >

The parameter word on the internal data bus 92 is written in the parameter register 52, in the step 613. In the step 614, it is determined whether or not the parameter is a parameter which accompanies a command which is to be executed in the execution unit 200. When it is determined YES in the step 614, the parameter word is written in the command and parameter buffer memory 210 in the execution unit 200, and the count PVC of the up-down counter 231 is incremented by one by the output PRWR of the AND circuit 86, in the step 615. In the construction of Fig. 16, the count CPVC of the up-down counter 239 is incremented by one by the output of the AND circuit 85'. When it is determined NO in the step 614, the step 615 is skipped. Then, in the step 618, the count of the basic parameter counter 62 is decremented by one in the step 618. When it is determined YES in the step 609, the operation goes to the step 624.

In the step 619, it is determined whether or not the output BPEND of the parameter counting portion 60 is ON ("1"). When it is determined NO, the operation goes to the step 611, and the next parameter word in the same basic parameter set is prefetched. When it is determined YES in the step 619, the count of the repeated number counter 64 is incremented in the step 620. In the step 621, it is determined whether or not the other output NREND of the parameter counting portion 60 is ON ("1"). When it is determined NO, the BP portion in the output of the command decoder 53 is loaded in the basic parameter counter 62 again in the step 622. When it is determined YES in the step 621, the operation goes to the step 624. In the step 623, it is determined whether or not the prefetch allowance signal WRRDY is ON ("1"). When it is determined YES, the operation goes to the step 611, and the next basic parameter set is prefetched. When it is determined NO in the step 623, the prefetch operation is stopped until the prefetch allowance signal WRRDY becomes "1", according to the aforementioned characteristic feature of the present invention.

In the step 624, it is determined whether or not the parameter is a parameter which accompanies a command which is to be executed in the execution unit 200. When it is determined No, e.g., the command is a branch command, the command is executed in the step 625. When it is determined YES, the operation goes to the step 626. In the step 627, the C/P flag register 84 is reset so that its $\overline{Q}$ output C/P becomes "1" by the micro ROM 71. In the step 628, it is determined whether or not the prefetch of a series of command sets is ended. The last word of each series of command sets contains a signal which indicates it is the end of the command set, and the micro ROM 71 carries out the above determination based on the signal.NO, the operation goes to the step 601, and the prefetch operation of the next command set is carried out. When it is determined YES, the operation ends.

(13) Control Operation in the Embodiment II

Figure 18 shows the control operation by the execution portion 224 in the embodiment of the present invention. The operation of Fig. 18 is written based on the construction of the execution unit interface 80 shown in Fig. 15A, but the operation of Fig. 18 is applicable to the construction of Fig. 16 when some portions are changed as explained below.

Initially, in the step 701, it is determined whether or not the count of the up-down counter 235 is equal to or more than one, i.e., the command and parameter buffer memory 210 holds a command word or a basic parameter set which is to be executed. When it is determined YES, it is determined whether or not the execution of data processing for a command and a basic parameter set which is held in the executing command set holding portion 220 is completed, in the step 702. When it is determined YES, a command or parameter word which is to be used in the next data processing operation in the the execution portion 224, is transferred from the command and parameter buffer memory 210 to the execution command set holding portion 220. At this time, when a command word is transferred in the step 703, the count CVC of the up-down counter 233 is decremented, and when a parameter word is transferred in the step 703, the count PVC of the up-down counter 231 is decremented. Or in the construction of Fig. 16, the count CPVC of the up-down counter 239 is decremented when either of a command word and a parameter word is transferred in the step 703. In the step 704, it is determined whether or not an amount of data (command and/or parameter) which is to be used in the next data processing operation in the execution portion 224, has been transferred. When it is determined NO, the operation goes to the step 703, and a next word is transferred. When it is determined YES in the step 704, the count EAC of the up-down counter 235 is decremented in the step 705. Next, in the steps 708 to 710, the command word held in the command code register 221 is decoded when it has not been decoded yet, and an execution of data processing operation for the command using a basic parameter set which is held in the executing basic parameter set register 223. In the step 711, it is determined whether or not the execution of data processing for the command set is completed. When it is determined NO, the operation goes to the step 701, or when it is

BNSDOCID: <EP___0389175A2_I_>

determined YES, the operation of the execution portion 224 ends.

(14) Example of Timing of Operation in the Construction of Fig. 16

Figure 19 shows an example of timing of the operation in the construction of Fig. 16. For a simple explanation, in Fig. 19, it is assumed that each basic parameter set is comprised of a parameter word.

In Fig. 19, C1, C2, C3 ... each denote a command word, and a corresponding execution period for a command set which begins with the command word. Pij (i = 1, 2, 3, ... , j = 1, 2, 3, ...) denotes a parameter word in a basic parameter set which follows the command word Ci (i = 1, 2, 3, ...).

It is assumed that, in the initial time t0, a command word C1 is prefetched in the command and parameter buffer memory (FIFO memory) 213. The count CPVC is incremented to "1", and as the command word C1 is not followed by a parameter word, the count EAC is incremented to "1".

In the time t1, the command word C1 is fetched into the execution portion 224, and the execution of the command word C1 is commenced. According to the above fetch of the word C1, both the counts of CPVC and EAC are decremented to "0".

In the time t2, the command word C2 which is to be followed by a basic parameter set comprised of a parameter word P21 only, is prefetched. Corresponding to the prefetch, the count CPVC is incremented to "1", and as the command word C1 is followed by a parameter word, the count EAC is not changed.

In the time t3, the above parameter word P21 which follows the above command word C2, is prefetched. Corresponding to the prefetch, both the counts CPVC and EAC are incremented to "1".

Similarly, the command word C3 is prefetched in the time t4, and the first basic parameter set P31 is prefetched in the time t5.

In the time t6, responding to the completion of the execution for the command word C1, the command word C2 and the basic parameter set P21 accompanying the command word C2, are fetched into the execution portion 224, and the execution for the command and parameter C2 and P21 is commenced.

The second basic parameter set P32 is prefetched in the time t7, the parameter word C4 is prefetched in the time t8, and the first basic parameter set P41 following the command word C4 is prefetched in the time t9.

When the second basic parameter set P42 which accompanies the command word C4 is prefetched in the time t10, the count CPVC becomes equal to the maximum number "6" of words which can be stored in the FIFO memory 213, and correspondingly, the output of the buffer vacancy detecting circuit 240 becomes active, and the Q output of the RS-flip-flop circuit 238, the prefetch allowance signal WRRDY becomes "0". Accordingly, the prefetch operation is stopped.

In the time t11, responding to the completion of the execution for the command word C2, the command word C3 and the basic parameter set P31 accompanying the command word C3, are fetched into the execution portion 224, and the execution for the command and parameter C3 and P31 is commenced. In the time t13, the basic parameter set P31 accompanying the command word C3, is fetched into the execution portion 224, and the execution for the command and parameter C3 and P32 is commenced.

In the time t15, responding to the completion of the execution for the command word C3, the command word C4 and the basic parameter set P41 accompanying the command word C4, are fetched into the execution portion 224, and the execution for the command and parameter C4 and P41 is commenced. At this time, the count EAC becomes "1", the output of the last operation amount detecting circuit 236 becomes active, and the RS-flip-flop circuit 238 is set to make its Q output active, i.e., the prefetch allowance signal WRRDY becomes active. Thus, the prefetch operation for a next word is restarted.

In the time t17, the basic parameter set P42 accompanying the command word C4, is fetched into the execution portion 224, and the execution for the command and parameter C4 and P42 is commenced. In the time t18, a next command word C5 is prefetched in the FIFO memory 213 by the above restart of the prefetch operation. After the time t8, operations similar to the above are carried out as shown in Fig. 19.

As understood from the example of Fig. 19, according to the present invention, the prefetch operations are continued as long as the capacity of the command and parameter buffer memory 213 allows the continuation, and is stopped as long as the execution portion 224 does not lack data which is to be executed. Thus, the frequency of the operations for requiring an allowance to newly use the system bus 505, and thus, the frequency of bus arbitrations, is minimized.

(15) Operations for Various Modes of Command Sets

Figures 20 to 23 show examples of various modes of the structures of command sets, other

than the structure shown in Fig. 7, where the operation of the parameter counting portion 60 for the command set having the structure shown in Fig. 7, is explained before in the item (9). The aforementioned MODE portion in each command word as shown in Fig. 10, indicates the above mode of the command set.

As explained before, the operation in the parameter counting portion 60 shown in Fig. 14, is changed in accordance with the MODE information. In the following, the operations of the parameter counting portion 60 shown in Fig. 14, for the various modes of command sets, are explained.

The mode of the command set having a structure shown in Fig. 20, is referred to as MODE II, while the mode of the command set having a structure shown in Fig. 7, is referred to as MODE I.

The command set having a structure shown in Fig. 20, is used in an image data transfer command for transferring image data through the image processor 500 to the graphic memory 503. As shown in Fig. 20, in the structure of Fig. 20, BP = r, and NP = n. In the image data transfer operation, the image data is transferred in parameter words, and the maximum amount of data which can be stored at one time in the command and parameter buffer memory 210, is transferred corresponding to one operation of the execution unit 200. In this example, the maximum amount is assumed to be five words. The above parameter words containing the maximum amount of data which can be stored at one time in the command and parameter buffer memory 210, is treated as the aforementioned basic parameter set, and thus, r = 5 is loaded in the basic parameter counter 62 as a preset value. The above number NP = n is loaded in the repeated number register 65 through the shifter 66 as a preset value. The gate circuit 61, the selector 63, and the shifter 66 are controlled by the aforementioned portion MODE in the content of the command word as shown in Fig. 10.

When the command set has a structure as shown in Fig. 20, the gate 61 is open, the selector 63 selects the aforementioned count signal CNT as its output, and the shifter does not shift the above number NP.

According to the above setting, the count of the basic parameter counter 62 indicates the number of parameter words which have not been prefetched yet in every five parameter words which are to be transferred. When all the parameter words have been prefetched, the basic parameter counter 62 outputs an active parameter zero signal.

The count of the repeated number counter 64 indicates the number of parameter words which have been already prefetched of all the parameter words which are to be transferred. The accord detecting circuit 79 compares the count of the repeated number counter 64 and the output of the repeated number register 65. When all the parameter words of all the parameter words have been prefetched, the accord detecting circuit 79 outputs a command set prefetch complete signal NPEND. The logical sum of the output of the basic parameter counter 62, and the output of the accord detecting circuit 79 is obtained in the OR circuit 67, and is output as a signal BPEND to the micro ROM 71 in the command fetch control portion 70 and to the execution unit interface 80. Thus, the BPEND signal becomes active every time a set of five parameter words are prefetched (a prefetch of a basic parameter set is completed), or when all the parameter words which are to be transferred, are prefetched.

The output of the repeated number counter 64 is also input into the FIFO memory 69. The FIFO memory 69 and the executing parameter number register 68 are provided to hold the number of parameter words which have been already prefetched of all the parameter words which are to be transferred, when an exceptional condition occurs. All the set of parameter words which have been prefetched are stored in the command and parameter buffer memory in the execution unit 200 until the image data in the set of parameter words are processed in the the execution unit 200. The set of parameter words which are stored in the command and parameter buffer memory in the execution unit 200 are at the same time held in the FIFO memory 69, and the number indicating the set of parameter words regarding which the data processing is executed in the execution unit 200, is held in the executing parameter number register 68. To realize the above, a signal PEXE which is output from the execution unit 200 when the execution unit 200 inputs a new set of parameter words to be executed, is applied to the FIFO memory 69 and the executing parameter number register 68, for an output control of the FIFO memory 69 and for an input control of the executing command register 68, respectively. Regarding the exception processing, a further explanation is given later.

The mode of the command set having a structure shown in Fig. 21, is referred to as MODE III.

The command set having a structure shown in Fig. 21, is used in a command for drawing a poli-line (a series of line segments where an end point of each line segment is connected to the beginning point of an adjacent line segment), or a command for drawing a trapezoid.

The structure of MODE III comprises a basic parameter set at the top, and structure of the other parameters following the basic parameter set is the same as the MODE II. For example, in the command for drawing a poli-line, the first basic parameter set contains two parameter words for drawing

BNSDOCID: <EP___0389175A2_I_>

a first line segment as shown in Fig. 8, but each of the other parameter words following the first basic parameter set, contains a coordinate of an end point of the next line segment. Therefore, the setting of the BP is changed after completion of the prefetch of the first basic parameter set.

The mode of the command set having a structure shown in Fig. 22, is referred to as MODE IV.

The command set having a structure shown in Fig. 22, is used in a command for transferring a predetermined number of parameter words, for example, to write values contained in the parameter words in corresponding internal registers, respectively. The data fetched by the command set having a structure of MODE IV, is not transferred to the execution unit 200.

For the operation of MODE IV, the count of the basic parameter counter 62 is not used. The selector 63 selects the aforementioned count signal CNT as its output, the shifter does not shift the above number NP, and therefore the above predetermined number NP = n is loaded in the repeated number register 65.

The count of the repeated number counter 64 indicates the number of parameter words which have been already transferred of all the parameter words which are to be transferred. The accord detecting circuit 79 compares the count of the repeated number counter 64 and the output of the repeated number register 65. When all the parameter words of all the parameter words have been transferred, the accord detecting circuit 79 outputs a command set prefetch complete signal NPEND.

The mode of the command set having a structure shown in Fig. 23, is referred to as MODE V. In the structure of MODE V, a plurality of parameters are contained in each parameter word. In the example of Fig. 23, each parameter word contains eight sets of four-bit parameter. The four-bit parameter is used for a called step command which indicates a relative displacement in the X-direction, +1, 0, or -1, and a relative displacement in the Y-direction, +1, 0, or -1, where two bits are used to indicate the displacement value +1, 0, or -1, in each of the X- and Y-directions.

When the command set has a structure as shown in Fig. 20, the gate 61 is open, the selector 63 selects the output of the basic parameter counter 62 as its output, and the shifter does not shift down the above number NP by the above number of parameter sets contained in each parameter word.

In the operation of MODE V, the parameter words are transferred to the execution unit 200 in basically the same way as the operation of MODE II. Namely, the the above four-bit parameter sets are transferred in the thirty-two bit parameter words, and the maximum amount of data which can be stored at one time in the command and parameter buffer memory 210, is transferred into the execution unit 200. In this example, the maximum amount is assumed to be five words. The above parameter words containing the maximum amount of data which can be stored at one time in the command and parameter buffer memory 210, is treated as the aforementioned basic parameter set, and thus, r = 5 is loaded in the basic parameter counter 62 as a preset value. The number NP = n which is contained in each command word, indicates the total number of the above four-bit sets of parameters. The number NP is eight-bit shifted through the shifter 66, and is loaded in the repeated number register 65 as a preset value. The gate circuit 61, the selector 63, and the shifter 66 are controlled by the aforementioned portion MODE in the content of the command word as shown in Fig. 10.

According to the above setting, the count of the basic parameter counter 62 indicates the number of parameter words which have not been prefetched yet in every five parameter words which are to be transferred. When all the parameter words have been prefetched, the basic parameter counter 62 outputs an active parameter zero signal.

The count of the repeated number counter 64 indicates the number of parameter words which have been already prefetched of all the parameter words which are to be transferred. The accord detecting circuit 79 compares the count of the repeated number counter 64 and the output of the repeated number register 65. When all the parameter words of all the parameter words have been prefetched, the accord detecting circuit 79 outputs a command set prefetch complete signal NPEND. The logical sum of the output of the basic parameter counter 62, and the output of the accord detecting circuit 79 is obtained in the OR circuit 67, and is output as a signal BPEND to the micro ROM 71 in the command fetch control portion 70 and to the execution unit interface 80. Thus, the BPEND signal becomes active every time a set of five parameter words are prefetched (a prefetch of a basic parameter set is completed), or when all the parameter words which are to be transferred, are prefetched.

The output of the repeated number counter 64 is also input into the FIFO memory 69. The FIFO memory 69 and the executing parameter number register 68 are provided to hold the number of parameter words which have been already prefetched of all the parameter words which are to be transferred, when an exceptional condition occurs. The set of parameter words which are stored in the command and parameter buffer memory in the execution unit 200 are at the same time held in the FIFO memory 69, and the number indicating

the set of parameter words regarding which the data processing is executed in the execution unit 200, is held in the executing parameter number register 68. To realize the above, a signal PEXE which is output from the execution unit 200 when the execution unit 200 inputs a new set of parameter words to be executed, is applied to the FIFO memory 69 and the executing parameter number register 68, for an output control of the FIFO memory 69 and for an input control of the executing command register 68, respectively. Regarding the exception processing, a further explanation is given later.

(16) Construction for Initializing Internal Registers

The aforementioned co-processors which are provided to take a part of the work of a central processor to enhance capability of the central processor, such as an image processor or an arithmetic processor, comprise internal registers which require initialization before executing data processing.

For example, in image processors, an interrupt mask register for setting a mask against an interrupt to the CPU, a refresh cycle register for defining a refresh cycle of a refresh memory wherein image data is stored, a stack pointer for storing an address to return when a subroutine is executed, a synchronizing signal pulse width register for storing a pulse width of horizontal and vertical synchronizing signals which are used in CRT display device, a back porch register for storing a back porch width, a cycle register for storing cycles of horizontal and vertical synchronizing signals, a current display width register for storing a width of a real space on the display plane, a virtual space register for storing a memory space wherein the data to be displayed is memorized, a split screen register for storing a location to display a split screen, a split screen address register for storing an address of a split screen, and the like, are internal registers which require initialization before executing data processing.

In the conventional image processor, the initialization of the internal registers is carried out under the control of the central processor in the system wherein the image processor is installed. Therefore, the load on the CPU at the initialization of the system is heavy, and total initialization time of the system is large.

Figure 24 shows an additional construction of the image processor 500 in the embodiment of the present invention.

In Fig. 24, the same reference numerals as explained before, each denote the same component as before. Reference numeral 11 denotes a DMA controller, 530 denotes an address register, 521 denotes an address field register, 522 denotes a selector, 523 denotes a decoder, and IR1, IR2, ... IRn each denote an internal register.

The internal registers IR1, IR2, ... IRn are respectively located in both the execution unit 200 and the display control unit 300 in the image processor 500.

In the construction of Fig. 24, the construction which is encircled by a dashed line 520 is additionally provided in the construction of Fig. 24.

The DMA controller 11 is contained in the access control circuit 10. The address register 530 is provided on the internal address bus 91, and when the CPU 501 accesses one of the internal registers IR1, IR2, ... IRn, the address signal of the internal register, which is supplied from the CPU 501 is latched in the address register 530, and the output of the address register 530 is applied to the decoder 523 through the selector 522. The decoder 523 decodes the applied address, and outputs an active input control signal to the corresponding internal register.

In initialization, the CPU 501 sends a start control signal to the DMA controller 11. Responding to the signal, the micro ROM 71 in the command fetch control portion 70 controls to fetch a predetermined address in the command and parameter memory 502 by controlling the DMA controller 11. The command word transferred from the command and parameter memory 502, is first latched in the command register 51, and is decoded in the command decoder 53. When it is determined that the command is not an internal register initialization command, the aforementioned prefetch operation is carried out.

When it is determined that the command is an internal register initialization command, the micro ROM 71 controls the address field register 521 to input the aforementioned portion REGISTER ADDRESS in the command word as shown in Fig. 10, through the internal data bus 92, and controls the selector 522 to select the output of the address field register 521. The portion REGISTER ADDRESS indicates the address of the internal register in which an initial value is to be written. The address which is applied to the decoder 523 through the selector 522, is decoded in the decoder 523, and an input control signal is output from the decoder 523 to the addressed internal register. The initial value is in advance written by the CPU 501 in the next address in the command and parameter memory 502. Therefore, the initial value is fetched into the parameter register 52 by the next fetch operation, and is supplied on the internal data bus 92. Thus, the initial value is written in the addressed internal register. Repeating operations similar to the above, all the internal

BNSDOCID: <EP___0389175A2_I_>

registers IR1, IR2, ... IRn are initialized.

(17) Exception Processing

When an exception arises in the central processor, the central processor carries out exception processing by itself. However, when an exception arises in a device other than the central processor, the exception is informed to the central processor by an interrupt processing. Responding to the information on the exception, the central processor carries out exception processing with regard to the device. In the exception processing, the central processor must know which command is in execution, and which parameter is in execution when the exception arose. Further, when the device is debugged, the command and parameter which are in execution, must be known.

In the embodiment of the present invention, as explained before with reference to Fig. 11, the command word which is executed in the execution unit 200 when an exceptional condition occurs, is held in the executing command register 56. As explained before with reference to Fig. 12, the address of the command word which is executed in the execution unit 200 when an exceptional condition occurs, is held in the executing command register 46. Further, as explained before with reference to Fig. 14, the number of the basic parameter set which is executed in the execution unit 200 when an exceptional condition occurs, is held in the executing parameter number register 68.

(18) Prefetches in Master Mode and Slave Mode

In the construction of Figs. 3A and 3B, it is assumed that the prefetch operations are carried out under the control of the micro ROM 71 using the DMA function of the access control circuit 10. The above prefetch operation under the control of the micro ROM 71 is called called master mode. Alternatively, in a called slave mode, the prefetch operations, in particular, the operations for starting and stopping prefetch based on the control signals generated in the control circuits provided in the execution unit 200, and the prefetch unit 100, can be carried out under the control of the CPU 501 using a similar construction to the explained above, while in the master mode, the prefetch operations are controlled by the micro ROM 71 as explained before. Namely, in the slave mode, the CPU 501 takes the place of the micro ROM 71 regarding the control of the prefetch operations.

(19) Application to Other Data Processing Systems

Although the above explanations are made for an image processing system comprising an image processor, as readily understood, all the characteristic features of the present invention do not directly relate to the executions of the commands in the execution portion 224 in Figs. 15A and 15B, or Fig. 16, except reading (fetching) of commands and parameters, and therefore, the present invention is applicable to all data processing systems comprising a data processing device for carrying out data processing other than the image data processing.

Claims

1. A data processing device (500) for carrying out data processing in accordance with one or more commands which are input from outside of said device (500), using one or more parameters which are input with respective commands, comprising a prefetch control means (100), an execution means and a data buffer means (210, 213) wherein

said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);

said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and

said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;

said data processing device (500) further comprises,

an insufficient space detecting means (60, 80, 231 - 235, 237, 238, 230) for detecting a first state of said data buffer means (210, 213) in which state there is not enough vacant space in the data buffer means (210, 213) in which a further command and/or a parameter can be stored, and

a least data detecting means (60, 80, 235, 236, 238) for detecting a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said execution means (200); and

said prefetch control means (100) starts said

BNSDOCID: <EP_ 0389175A2 I >

prefetch operation and continues successive prefetch operations when said second state is detected, and stops the prefetch operation when said first state is detected.

2. A data processing system comprising said data processing device (500) according to claim 1; further comprising a command and parameter memory (502) for storing said commands and parameters in the order in which order the commands are executed and the parameters are used in the execution means (200).

3. A data processing device (500) for carrying out data processing in accordance with one or more commands which are input from outside of said device (500), using one or more parameters which are input with respective commands, comprising a prefetch control means (100), an execution means (200) and a data buffer means (210, 213), wherein

said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);

said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and

said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;

said data processing device (500) further comprises,

an insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) for detecting a first state of said data buffer means (210, 213) in which state there is not enough vacant space in the data buffer means (210, 213) in which a further command and/or a parameter can be stored, and

a least data detecting means (60, 80, 235, 236, 238) for detecting a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said execution (200); and

said prefetch control means (100) starts said prefetch operation and continues successive prefetch operations when said second state is detected, and stops the prefetch operation when said first state is detected, and

said prefetch control means (100) comprises a branch command detecting means (51, 53) for detecting a branch command in prefetched commands, and said prefetch control means (100) stops the prefetch operation until a result of the execution of said branch command is obtained in said execution means (200) when a branch command is detected.

4. A data processing system comprising said data processing device (500) according to claim 3, further comprising a command and parameter memory (502) for storing said commands and parameters in the order in which order the commands are executed and the parameters are used in the execution means (200).

5. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and the data processing device (500) according to claim 1; wherein

said command and parameter memory (502) stores said commands and parameters in the order the commands are executed and the parameters are used in the data processing device (500);

when successive executions of a command by said data processing device (500) are required, which executions each use a parameter set comprised of one or more parameters, the command is stored in said command and parameter memory (502), with followed by the parameter sets which are respectively used in successive executions of the command; and

said command contains information on the number of said parameter sets, and information on the data length of each of the parameter sets.

6. A data processing device (500) used in said data processing system according to claim 5, wherein

said data processing device (500) further comprises,

a command decode means (51, 53) for detecting said number of the parameter sets and the data length of each of the parameter sets from a prefetched command;

a data length counting means (62, 68) for counting the length of prefetched data to detect an end of each of prefetched parameter sets; and

a parameter set number counting means (63, 64, 65, 79) for counting the number of parameter sets which are prefetched into said data buffer (210, 213) to determine whether a command is prefetched or a parameter set is prefetched, and determine whether or not all the parameter sets following a command is prefetched.

7. A data processing device (500) according to claim 6, wherein

said data processing device (500) further com-

BNSDOCID: <EP___0389175A2_I_>

prises a parameter number FIFO means (69) and an executing parameter number register means (68),

said parameter number FIFO means (69) inputs the count of said parameter set number counting means (63, 64, 65, 79), and outputs its own oldest content into said executing parameter number register means (68) to hold a number indicating the parameter set which is currently in execution, where the content of said executing parameter number register means (69) is renewed responding to an input of a new parameter set into said execution means (220).

8. A data processing device (500) according to claim 6, wherein

said data buffer (210, 213) in said data processing device (500) comprises a FIFO memory (213) into which prefetched commands and parameters are input in the order of the prefetch.

9. A data processing device (500) according to claim 8, wherein

said insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) increments a count responding to a prefetch of every unit length of commands and parameters for said detection, and decrements the count responding to every input of every unit length of commands and parameters into said execution means (220); and

said least data detecting means (60, 80, 235, 236, 238) increments a count when a command which is not followed by a parameter is prefetched, and when every parameter set is prefetched, and decrements the count when a command which is not followed by a parameter is input into said execution means (220); and when every parameter set is input into said execution means (220).

10. A data processing device (500) according to claim 6, wherein

said data buffer (210, 213) in said data processing device (500) comprises a command buffer means (211) for storing said prefetched commands, and a parameter buffer means (212) for storing said prefetched parameters; and

said data processing device (500) further comprises a command buffer input control means (71, 84, 81, 85) for controlling input of a prefetched command in said command buffer means (211) based on the output of said parameter set number counting means (63, 64, 65, 79), and a parameter buffer input control means (71, 84, 82, 83, 86) for controlling input of a prefetched parameter set in said parameter buffer means (212) based on the outputs of said data length counting means (62, 68) and said parameter set number counting means (63, 64, 65, 79).

11. A data processing device (500) according to claim 10, wherein

said insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) comprises,

a command buffer amount counter means (233) for incrementing a count responding to every prefetch of a command, decrementing the count responding to every input of a command into said execution means (220), and detecting a third state wherein not enough vacant space exists in the command buffer means (211) in which a further command can be stored,

a parameter buffer amount counter means (231) for incrementing a count responding to every prefetch of a unit length of parameters, and decrementing the count responding to every input of a unit length of parameters into said execution means (220), and detecting a fourth state wherein not enough vacant space exists in the command buffer means (212) in which a further command can be stored,

a logical OR means for detecting either of said third and fourth states; and

said data processing device (500) further comprises a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said executing; and

said least data detecting means (60, 80, 235, 236, 238) increments a count when a command which is not followed by a parameter is prefetched, and when every parameter set is prefetched, and decrements the count when a command which is not followed by a parameter is input into said execution means (220), and when every parameter set is input into said execution means (220).

12. A data processing device (500) according to claim 6, wherein

said prefetch control means (100) comprises an address calculation means (40) for calculating an address for prefetching a command or a parameter, and

said address calculation means (40) comprises a command address FIFO means (45) and an executing command address register means (46),

said command address FIFO means (46) inputs the output of said address calculation means (40), and outputs its own oldest content into said executing command address register means (45) to hold the address of a command which is currently in execution, where the content of said executing command address register means (46) is renewed responding to an input of a new command into said execution means (220).

13. A data processing device (500) according to claim 6, wherein

said prefetch control means (100) comprises a command register means (51) for temporarily holding a prefetched command, and

said command register means (51) comprises a command FIFO means (55) and an executing com-

EP0389175A2 I

mand register means (56),
said command FIFO means (55) inputs the output of said command register means (51), and outputs its own oldest content into said executing command register means (56) to hold a command which is currently in execution, where the content of said executing command register means (56) is renewed responding to an input of a new command into said execution means (220).

14. A data processing system according to claim 5, wherein
said command further contains information on the format of said parameter sets, and information on the data length of each of the parameter sets.

15. A data processing device (500) used in said data processing system according to claim 14, wherein
said data processing device (500) further comprises,
said command decode means (51, 53) further detects said format of said parameter sets;
said data length counting means (62, 68) can count the length of prefetched data further to detect a length of prefetched parameter sets corresponding to a maximum capacity for parameters which can be stored in said data buffer (210, 213) according to said format.

16. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and the data processing device (500) according to claim 1; wherein
said command and parameter memory (502) stores said commands and parameters in the order the commands are executed and the parameters are used in the data processing device (500);
said data processing device (500) further comprises one or more internal registers which are requested to be initialized;
one or more initialization commands can be stored in said command and parameter memory (502), where said initialization commands respectively contain internal addresses of said internal registers; and
said initialization commands each can be followed by parameter data which contains initialization data which is to be written in a corresponding internal register, in said command and parameter memory (502).

17. A data processing device (500) used in said data processing system according to claim 16, wherein
said prefetch control means (100) comprises,
a command decode means (51, 53) for determining whether or not a command fetched from said command and parameter memory (502) is said initialization command;
a parameter register means (52) for receiving said initialization data therein; and
an address field register means (521) for holding said internal address contained in the initialization command;
an address decoder means (523) for decoding its own input to output an input control signal to the internal register which is addressed by the internal address;
a switching means (522) for outputting either of an address signal supplied from outside of the data processing device (500), and the output of said address field register means (521); and
an initialization control means for controlling said switching means (522) to select the output of said address field register means (521), as its own output, when an initialization command is determined.

18. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and the data processing device (500) according to claim 1; wherein
said execution means (200) having a first port, connected to a first bus (505) through said prefetch control means (100), for inputting said commands and parameters, and a second port, connected to a second bus (506), for outputting results of the execution.

19. A data processing device (500) for carrying out data processing in accordance with one or more commands which are input from outside of said device (500), using one or more parameters which are input with respective commands, comprising a prefetch control means (100), an execution means (200) and a data buffer means (210, 213), wherein
said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);
said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and
said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;
said data processing device (500) further com-

BNSDOCID: <EP___0389175A2_I_>

prises,
an insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) for detecting a first state of said data buffer means (210, 213) in which state there is not enough vacant space in the data buffer means (210, 213) in which a further command and/or a parameter can be stored, and
a least data detecting means (60, 80, 235, 236, 238) for detecting a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said execution means (200); and
said prefetch control means (100) comprises a branch command detecting (51, 53) means for detecting a branch command in prefetched commands, and said prefetch control means (100) stops the prefetch operation until a result of the execution of said branch command is obtained in said execution means (200) when a branch command is detected.

20. A data processing system comprising said data processing device (500) according to claim 19, further comprising a command and parameter memory (502) for storing said commands and parameters in the order in which order the commands are executed and the parameters are used in the execution means (200).

21. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and a data processing device (500); wherein
said data processing device (500) inputs said commands and parameters, carries out data processing in accordance with the commands which are input from outside of said device (500), using said parameters, comprising a prefetch control means (100), an execution means (200) and a data buffer means (210, 213), wherein
said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);
said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and
said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;
when successive executions of a command by said data processing device (500) are required, which executions each use a parameter set comprised of one or more parameters, the command is stored in said command and parameter memory (502), with followed by the parameter sets which are respectively used in successive executions of the command; and
said command contains information on the number of said parameter sets, and information on the data length of each of the parameter sets.

22. A data processing device (500) used in said data processing system according to claim 21, wherein
said data processing device (500) further comprises,
a command decode means (51, 53) for detecting said number of the parameter sets and the data length of each of the parameter sets from a prefetched command;
a data length counting means (62, 68) for counting the length of prefetched data to detect an end of each of prefetched parameter sets; and
a parameter set number counting means (63, 64, 65, 79) for counting the number of parameter sets which are prefetched into said data buffer (210, 213) to determine whether a command is prefetched or a parameter set is prefetched, and determine whether or not all the parameter sets following a command is prefetched.

23. A data processing device (500) according to claim 22, wherein
said data processing device (500) further comprises a parameter number FIFO means (69) and an executing parameter number register means (68),
said parameter number FIFO means (69) inputs the count of said parameter set number counting means (63, 64, 65, 79), and outputs its own oldest content into said executing parameter number register means (68) to hold a number indicating the parameter set which is currently in execution, where the content of said executing parameter number register means (69) is renewed responding to an input of a new parameter set into said execution means (220).

24. A data processing device (500) according to claim 22, wherein
said data buffer (210, 213) in said data processing device (500) comprises a FIFO memory (213) into which prefetched commands and parameters are input in the order of the prefetch.

25. A data processing device (500) according to claim 22, wherein
said data buffer (210, 213) in said data processing device (500) comprises a command buffer means

BNSDOCID: <EP 0389175A2 >

(211) for storing said prefetched commands, and a parameter buffer means (212) for storing said prefetched parameters;

said data processing device (500) further comprises a command buffer input control means (71, 84, 81; 85) for controlling input of a prefetched command in said command buffer means (211) based on the output of said parameter set number counting means (63, 64, 65, 79), and a parameter buffer input control means (71, 84, 82, 83, 86) for controlling input of a prefetched parameter set in said parameter buffer means (212) based on the outputs of said data length counting means (62, 68) and said parameter set number counting means (63, 64, 65, 79).

26. A data processing device (500) according to claim 22, wherein

said prefetch control means (100) comprises an address calculation means (40) for calculating an address for prefetching a command or a parameter, and

said address calculation means (40) comprises a command address FIFO means (45) and an executing command address register means (46),

said command address FIFO means (46) inputs the output of said address calculation means (40), and outputs its own oldest content into said executing command address register means (45) to hold the address of a command which is currently in execution, where the content of said executing command address register means (46) is renewed responding to an input of a new command into said execution means (220).

27. A data processing device (500) according to claim 22, wherein

said prefetch control means (100) comprises a command register means (51) for temporarily holding a prefetched command, and

said command register means (51) comprises a command FIFO means (55) and an executing command register means (56),

said command FIFO means (55) inputs the output of said command register means (51), and outputs its own oldest content into said executing command register means (56) to hold a command which is currently in execution, where the content of said executing command register means (56) is renewed responding to an input of a new command into said execution means (220).

28. A data processing system according to claim 21, wherein

said command further contains information on the format of said parameter sets, and information on the data length of each of the parameter sets.

29. A data processing device (500) used in said data processing system according to claim 28, wherein

in said data processing device (500),

said command decode means (51, 53) further detects said format of said parameter sets;

said data length counting means (62, 68) can count the length of prefetched data further to detect a length of prefetched parameter sets corresponding to a maximum capacity for parameters which can be stored in said data buffer (210, 213) according to said format.

30. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and a data processing device (500); wherein

said data processing device (500) inputs said commands and parameters, carries out data processing in accordance with the commands which are input from outside of said device (500), using said parameters, comprising a prefetch control means (100), an execution means (200) and a data buffer means (210, 213), wherein

said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);

said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and

said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;

said command and parameter memory (502) stores said commands and parameters in the order the commands are executed and the parameters are used in the data processing device (500);

said data processing device (500) further comprises one or more internal registers which are requested to be initialized;

one or more initialization commands can be stored in said command and parameter memory (502), where said initialization commands respectively contain internal addresses of said internal registers; and

said initialization commands each can be followed by parameter data which contains initialization data which is to be written in a corresponding internal register, in said command and parameter memory (502).

31. A data processing device (500) used in said data processing system according to claim 30,

BNSDOCID: <EP___0389175A2_I_>

wherein
said prefetch control means (100) comprises,
a command decode means (51, 53) for determining whether or not a command fetched from said command and parameter memory (502) is said initialization command;
a parameter register means (52) for receiving said initialization data therein; and
an address field register means (521) for holding said internal address contained in the initialization command;
an address decoder means (523) for decoding its own input to output an input control signal to the internal register which is addressed by the internal address;
a switching means (522) for outputting either of an address signal supplied from outside of the data processing device (500), and the output of said address field register means (521); and
an initialization control means for controlling said switching means (522) to select the output of said address field register means (521), as its own output, when an initialization command is determined.

32. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and a data processing device (500); wherein
said data processing device (500) inputs said commands and parameters, carries out data processing in accordance with the commands which are input from outside of said device (500), using said parameters, comprising a prefetch control means (100), an execution means (200) and a data buffer means (210, 213), wherein
said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);
said prefetch control means (100) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and
said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;
said execution means (200) having a first port, connected to a first bus (505) through said prefetch control means (100), for inputting said commands and parameters, and a second port, connected to a second bus (506), for outputting results of the execution.

33. A data processing device (500) for carrying out data processing in accordance with one or more commands which are input from outside of said device (500), using one or more parameters which are input with respective commands, comprising an execution means (200) and a data-buffer means (210, 213), wherein
said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200); and
said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;
said data processing device (500) further comprises,
an insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) for detecting a first state of said data buffer means (210, 213) in which state there is not enough vacant space in the data buffer means (210, 213) in which a further command and/or a parameter can be stored; and
a least data detecting means (60, 80, 235, 236, 238) for detecting a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said execution means (200).

34. A data processing system comprising said data processing device (500) according to claim 33, further comprising,
a command and parameter memory (502) for storing said commands and parameters in the order in which order the commands are executed and the parameters are used in the execution means (200), and
a prefetch control means (501), provided outside of said data processing device (500), controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213), and
said prefetch control means (501) starts said prefetch operation and continues successive prefetch operations when said second state is detected, and stops the prefetch operation when said first state is detected.

35. A data processing system comprising said data processing device (500) according to claim

34. wherein said prefetch control means (501) comprises a branch command detecting means (51, 53) for detecting a branch command in prefetched commands, and said prefetch control means (501), and further stops the prefetch operation until a result of the execution of said branch command is obtained in said execution means (200).

36. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a prefetch control means (501), a command and parameter memory (502), and the data processing device (500) according to claim 33; wherein

said command and parameter memory (502) stores said commands and parameters in the order the commands are executed and the parameters are used in the data processing device (500);

said prefetch control means (501), provided outside of said data processing device (500), controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213), and said prefetch control means (501) starts said prefetch operation and continues successive prefetch operations when said second state is detected, and stops the prefetch operation when said first state is detected;

when successive executions of a command by said data processing device (500) are required, which executions each use a parameter set comprised of one or more parameters, the command is stored in said command and parameter memory (502), with followed by the parameter sets which are respectively used in successive executions of the command; and

said command contains information on the number of said parameter sets, and information on the data length of each of the parameter sets.

37. A data processing device (500) used in said data processing system according to claim 33, wherein

said data processing device (500) further comprises,

a command decode means (51, 53) for detecting said number of the parameter sets and the data length of each of the parameter sets from a prefetched command;

a data length counting means (62, 68) for counting the length of prefetched data to detect an end of each of prefetched parameter sets; and

a parameter set number counting means (63, 64, 65, 79) for counting the number of parameter sets which are prefetched into said data buffer (210, 213) to determine whether a command is prefetched or a parameter set is prefetched, and determine whether or not all the parameter sets following a command is prefetched.

38. A data processing device (500) according to claim 37, wherein

said data processing device (500) further comprises a parameter number FIFO means (69) and an executing parameter number register means (68),

said parameter number FIFO means (69) inputs the count of said parameter set number counting means (63, 64, 65, 79), and outputs its own oldest content into said executing parameter number register means (68) to hold a number indicating the parameter set which is currently in execution, where the content of said executing parameter number register means (69) is renewed responding to an input of a new parameter set into said execution means (220).

39. A data processing device (500) according to claim 37, wherein

said data buffer (210, 213) in said data processing device (500) comprises a FIFO memory (213) into which prefetched commands and parameters are input in the order of the prefetch.

40. A data processing device (500) according to claim 39, wherein

said insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) increments a count responding to a prefetch of every unit length of commands and parameters for said detection, and decrements the count responding to every input of every unit length of commands and parameters into said execution means (220); and

said least data detecting means (60, 80, 235, 236, 238) increments a count when a command which is not followed by a parameter is prefetched, and when every parameter set is prefetched; and decrements the count when a command which is not followed by a parameter is input into said execution means (220), and when every parameter set is input into said execution means (220).

41. A data processing device (500) according to claim 37, wherein

said data buffer (210, 213) in said data processing device (500) comprises a command buffer means (211) for storing said prefetched commands, and a parameter buffer means (212) for storing said prefetched parameters;

said data processing device (500) further comprises a command buffer input control means (71, 84, 81, 85) for controlling input of a prefetched command in said command buffer means (211) based on the output of said parameter set number counting means (63, 64, 65, 79), and a parameter buffer input control means (71, 84, 82, 83, 86) for controlling input of a prefetched parameter set in said parameter buffer means (212) based on the outputs of said data length counting means (62, 68) and said parameter set number counting means (63, 64, 65, 79).

BNSDOCID: <EP___0389175A2_I_>

42. A data processing device (500) according to claim 41, wherein
said insufficient space detecting means (60, 80, 231 - 235, 237, 238, 239) comprises,
a command buffer amount counter means (233) for incrementing a count responding to every prefetch of a command, decrementing the count responding to every input of a command into said execution means (220), and detecting a third state wherein not enough vacant space exists in the command buffer means (211) in which a further command can be stored,
a parameter buffer amount counter means (231) for incrementing a count responding to every prefetch of a unit length of parameters, and decrementing the count responding to every input of a unit length of parameters into said execution means (220), and detecting a fourth state wherein not enough vacant space exists in the command buffer means (212) in which a further command can be stored
a logical OR means for detecting either of said third and fourth states; and
said data processing device (500) further comprises a second state of said data buffer means (210, 213) in which state the data buffer means (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in said executing; and
said least data detecting means (60, 80, 235, 236, 238) increments a count when a command which is not followed by a parameter is prefetched, and when every parameter set is prefetched, and decrements the count when a command which is not followed by a parameter is input into said execution means (220), and when every parameter set is input into said execution means (220).

43. A data processing system according to claim 34, wherein
said prefetch control means (501) comprises an address calculation means (40) for calculating an address for prefetching a command or a parameter, and
said address calculation means (40) comprises a command address FIFO means (45) and an executing command address register means (46),
said command address FIFO means (46) inputs the output of said address calculation means (40), and outputs its own oldest content into said executing command address register means (45) to hold the address of a command which is currently in execution, where the content of said executing command address register means (46) is renewed responding to an input of a new command into said execution means (220).

44. A data processing system according to claim 33, wherein
said prefetch control means (501) comprises a command register means (51) for temporarily holding a prefetched command, and
said command register means (51) comprises a command FIFO means (55) and an executing command register means (56),
said command FIFO means (55) inputs the output of said command register means (51), and outputs its own oldest content into said executing command register means (56) to hold a command which is currently in execution, where the content of said executing command register means (56) is renewed responding to an input of a new command into said execution means (220).

45. A data processing system according to claim 34, wherein
said command further contains information on the format of said parameter sets, and information on the data length of each of the parameter sets.

46. A data processing device (500) used in said data processing system according to claim 45, wherein
said data processing device (500) further comprises,
said command decode means (51, 53) further detects said format of said parameter sets;
said data length counting means (62, 68) can count the length of prefetched data further to detect a length of prefetched parameter sets corresponding to a maximum capacity for parameters which can be stored in said data buffer (210, 213) according to said format.

47. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and the data processing device (500) according to claim 33; wherein
said execution means (200) having a first port, connected to a first bus (505) through said prefetch control means (501), for inputting said commands and parameters, and a second port, connected to a second bus (506), for outputting results of the execution.

48. A data processing system comprising said data processing device (500) according to claim 33, further comprising a command and parameter memory (502) for storing said commands and parameters in the order in which order the commands are executed and the parameters are used in the execution means (200);
said prefetch control means (501) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and
said prefetch control means (501) comprises a branch command detecting (51, 53) means for detecting a branch command in prefetched commands, and said prefetch control means (501)

EPODOC: EP 0389175A2 1

stops the prefetch operation until a result of the execution of said branch command is obtained in said execution means (200).

49. A data processing system for carrying out data processing in accordance with one or more commands, using one or more parameters accompanying respective commands, comprising a command and parameter memory (502) and a data processing device (500); wherein

said data processing device (500) inputs said commands and parameters, carries out data processing in accordance with the commands which are input from outside of said device (500), using said parameters, comprising a prefetch control means (501), an execution means (200) and a data buffer means (210, 213), wherein

said data buffer means (210, 213) for storing one or more commands and/or one or more parameters which are prefetched, until each of said commands and parameters is read out by said execution means (200);

said prefetch control means (501) controls a prefetch operation of data including a command and/or a parameter from outside of said device (500) into said data buffer means (210, 213); and

said execution means (200) inputs a command which is the oldest one of the commands stored in said data buffer means (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in said data buffer means (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein;

when successive executions of a command by said data processing device (500) are required, which executions each use a parameter set comprised of one or more parameters, the command is stored in said command and parameter memory (502), with followed by the parameter sets which are respectively used in successive executions of the command; and

said command contains information on the number of said parameter sets, and information on the data length of each of the parameter sets.

50. A data processing device (500) used in said data processing system according to claim 49, wherein

said data processing device (500) further comprises,

a command decode means (51, 53) for detecting said number of the parameter sets and the data length of each of the parameter sets from a prefetched command;

a data length counting means (62, 68) for counting the length of prefetched data to detect an end of each of prefetched parameter sets; and

a parameter set number counting means (63, 64, 65, 79) for counting the number of parameter sets which are prefetched into said data buffer (210, 213) to determine whether a command is prefetched or a parameter set is prefetched, and determine whether or not all the parameter sets following a command is prefetched.

51. A data processing device (500) according to claim 50, wherein

said data processing device (500) further comprises a parameter number FIFO means (69) and an executing parameter number register means (68),

said parameter number FIFO means (69) inputs the count of said parameter set number counting means (63, 64, 65, 79), and outputs its own oldest content into said executing parameter number register means (68) to hold a number indicating the parameter set which is currently in execution, where the content of said executing parameter number register means (69) is renewed responding to an input of a new parameter set into said execution means (220).

52. A data processing device (500) according to claim 50, wherein

said data buffer (210, 213) in said data processing device (500) comprises a FIFO memory (213) into which prefetched commands and parameters are input in the order of the prefetch.

53. A data processing device (500) according to claim 50, wherein

said data buffer (210, 213) in said data processing device (500) comprises a command buffer means (211) for storing said prefetched commands, and a parameter buffer means (212) for storing said prefetched parameters;

said data processing device (500) further comprises a command buffer input control means (71, 84, 81, 85) for controlling input of a prefetched command in said command buffer means (211) based on the output of said parameter set number counting means (63, 64, 65, 79), and a parameter buffer input control means (71, 84, 82, 83, 86) for controlling input of a prefetched parameter set in said parameter buffer means (212) based on the outputs of said data length counting means (62, 68) and said parameter set number counting means (63, 64, 65, 79).

54. A data processing system according to claim 49, wherein

said prefetch control means (501) comprises an address calculation means (40) for calculating an address for prefetching a command or a parameter, and

said address calculation means (40) comprises a command address FIFO means (45) and an executing command address register means (46),

said command address FIFO means (46) inputs the output of said address calculation means (40), and

BNSDOCID: <EP___0389175A2_I_>

outputs its own oldest content into said executing command address register means (45) to hold the address of a command which is currently in execution, where the content of said executing command address register means (46) is renewed responding to an input of a new command into said execution means (220).

55. A data processing system according to claim 49, wherein
said prefetch control means (501) comprises a command register means (51) for temporarily holding a prefetched command, and
said command register means (51) comprises a command FIFO means (55) and an executing command register means (56),
said command FIFO means (55) inputs the output of said command register means (51), and outputs its own oldest content into said executing command register means (56) to hold a command which is currently in execution, where the content of said executing command register means (56) is renewed responding to an input of a new command into said execution means (220).

56. A data processing system according to claim 49, wherein
said command further contains information on the format of said parameter sets, and information on the data length of each of the parameter sets.

57. A data processing device (500) used in said data processing system according to claim 56, wherein
in said data processing device (500),
said command decode means (51, 53) further detects said format of said parameter sets;
said data length counting means (62, 68) can count the length of prefetched data further to detect a length of prefetched parameter sets corresponding to a maximum capacity for parameters which can be stored in said data buffer (210, 213) according to said format.

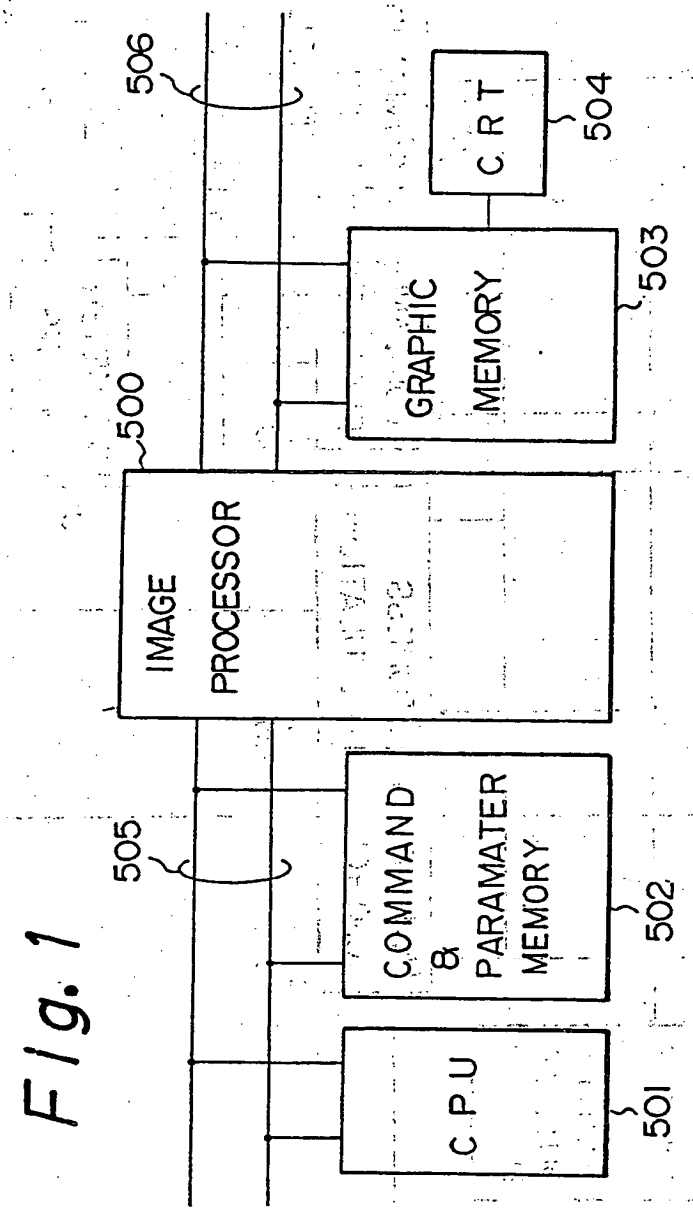


Fig. 1

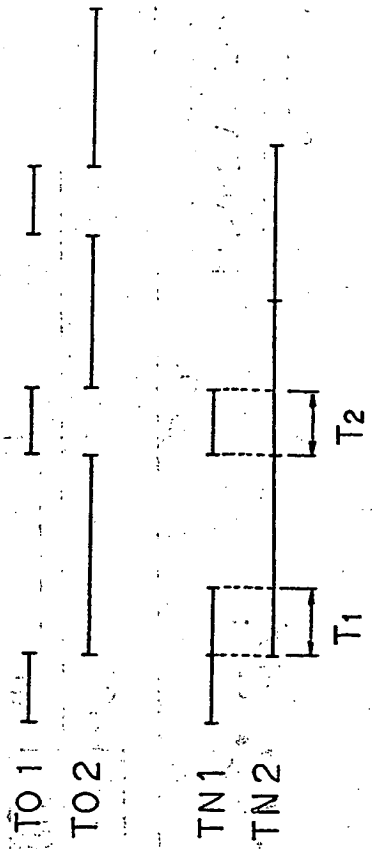


Fig. 2

BNSDOCID: <EP___0389175A2_I_>

Fig. 3

Fig. 3A	Fig. 3B

Fig. 3A

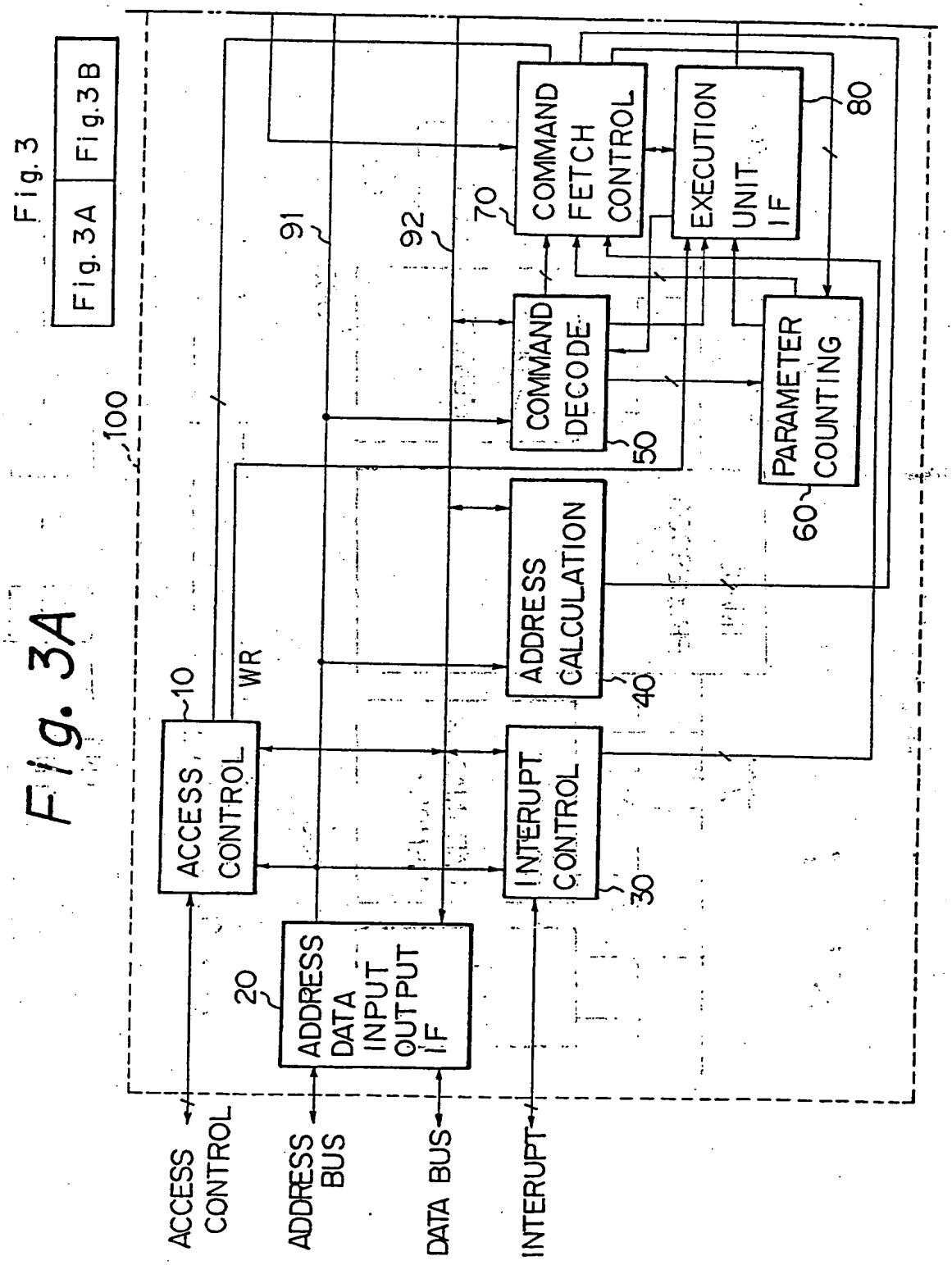
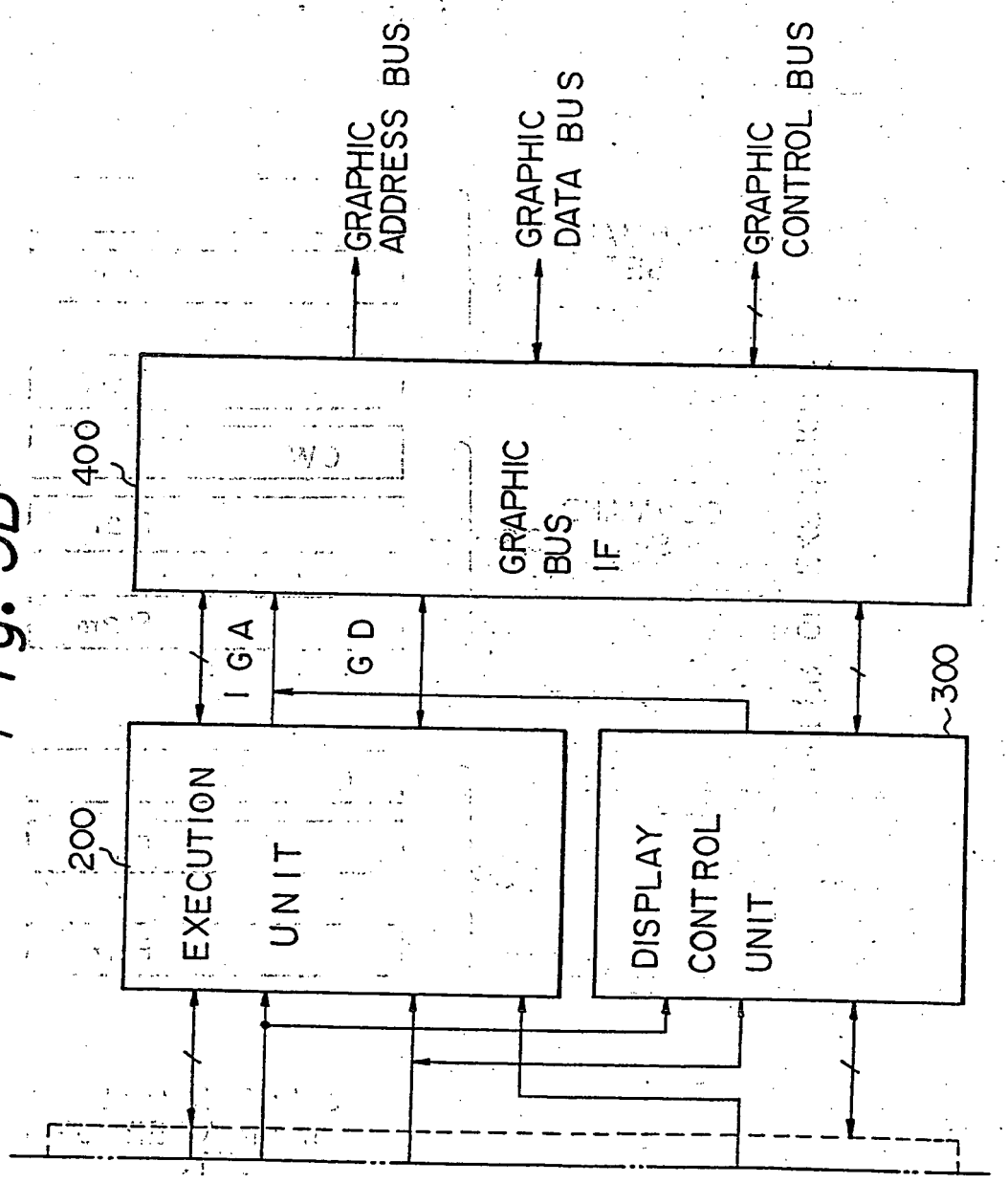
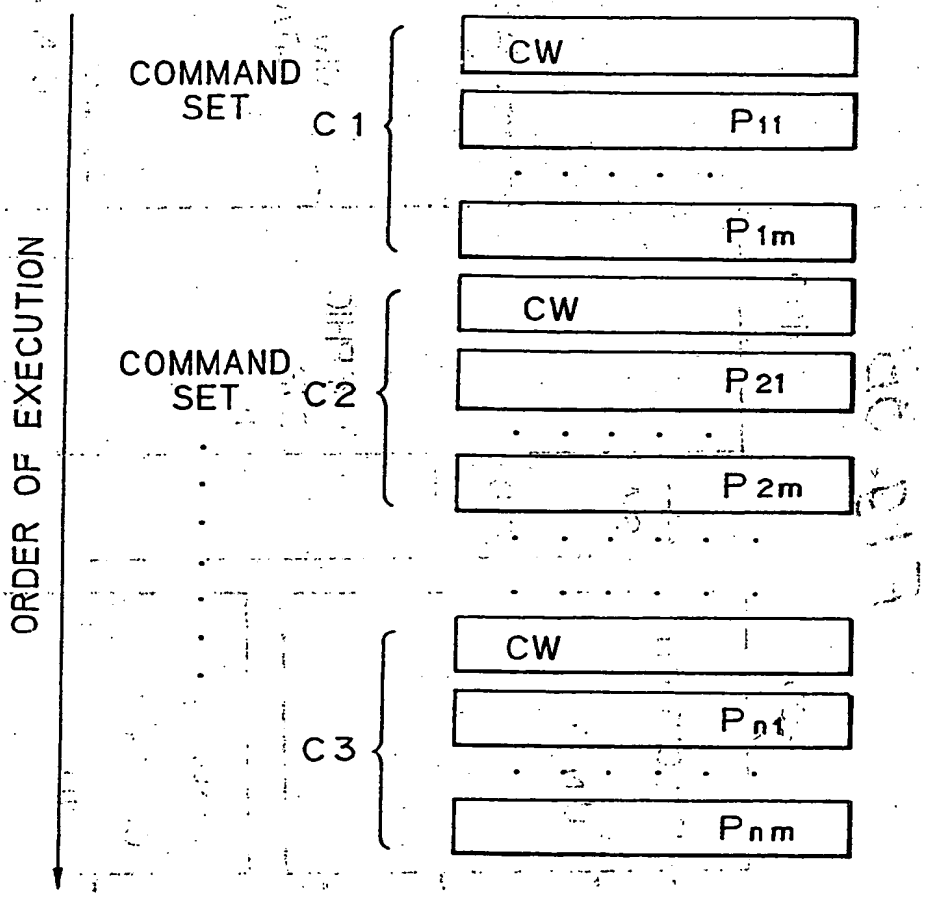


Fig. 3B



BNSDOCID: <EP___0389175A2_I_>

Fig. 4



CW : COMMAND
n : NUMBER OF COMMAND SETS
m : NUMBER OF PARAMETER WORDS IN A COMMAND SET
$P_{11} \sim nm$: PARAMETER WORDS

Fig. 5

0000	LINE SEGMENT GENERATING COMMAND
0001	STARTING POINT (X_{S0}, Y_{S0})
0010	ENDING POINT (X_{E0}, Y_{E0})
0011	LINE SEGMENT GENERATING COMMAND
0100	STARTING POINT (X_{S1}, Y_{S1})
0101	ENDING POINT (X_{E1}, Y_{E1})
0110	LINE SEGMENT GENERATING COMMAND
0111	STARTING POINT (X_{S3}, Y_{S3})
1000	ENDING POINT (X_{E3}, Y_{E3})
1001	LINE SEGMENT GENERATING COMMAND
1010	STARTING POINT (X_{S4}, Y_{S4})
1011	ENDING POINT (X_{E4}, Y_{E4})

BNSDOCID: <EP___0389175A2_I_>

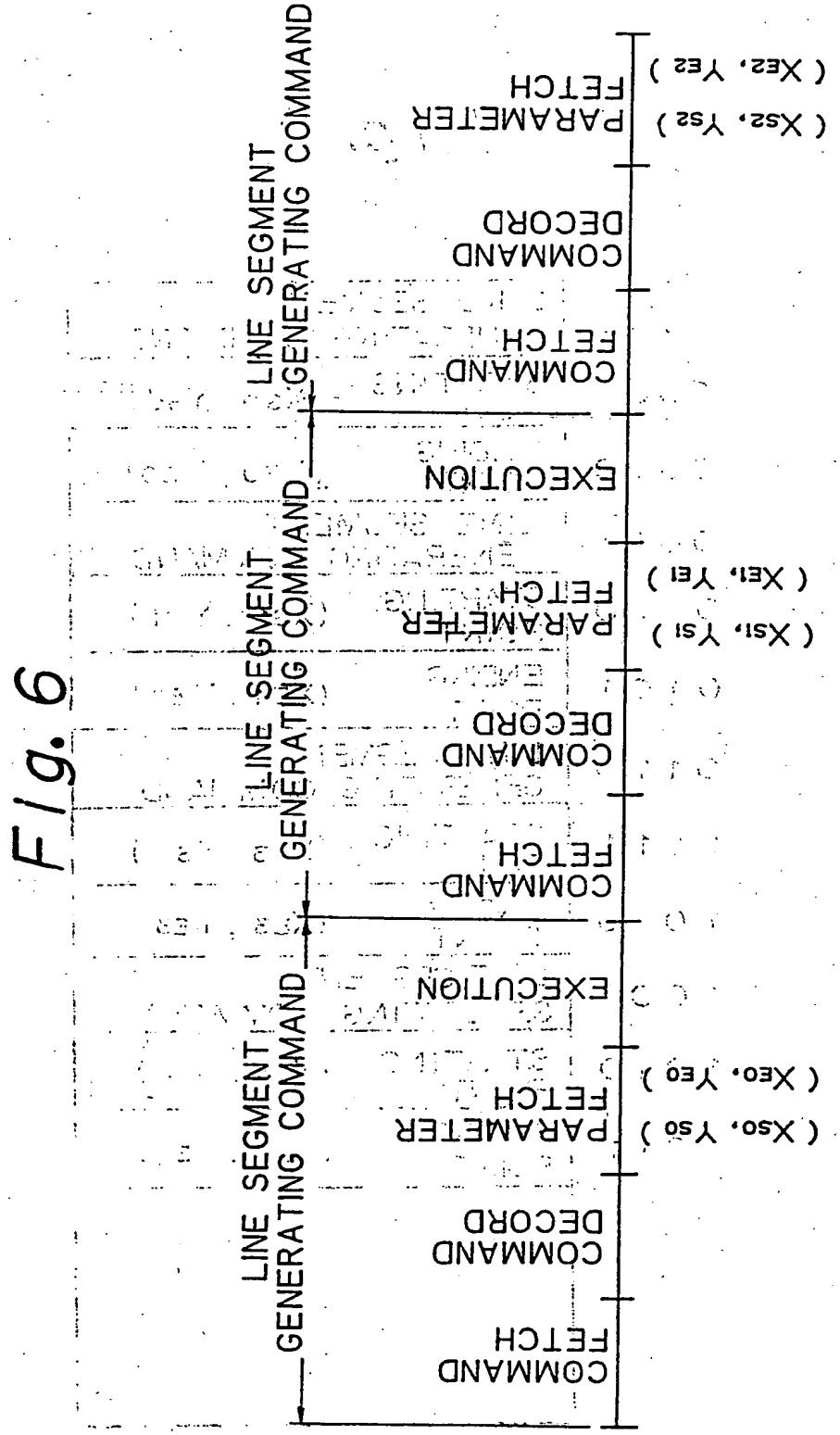
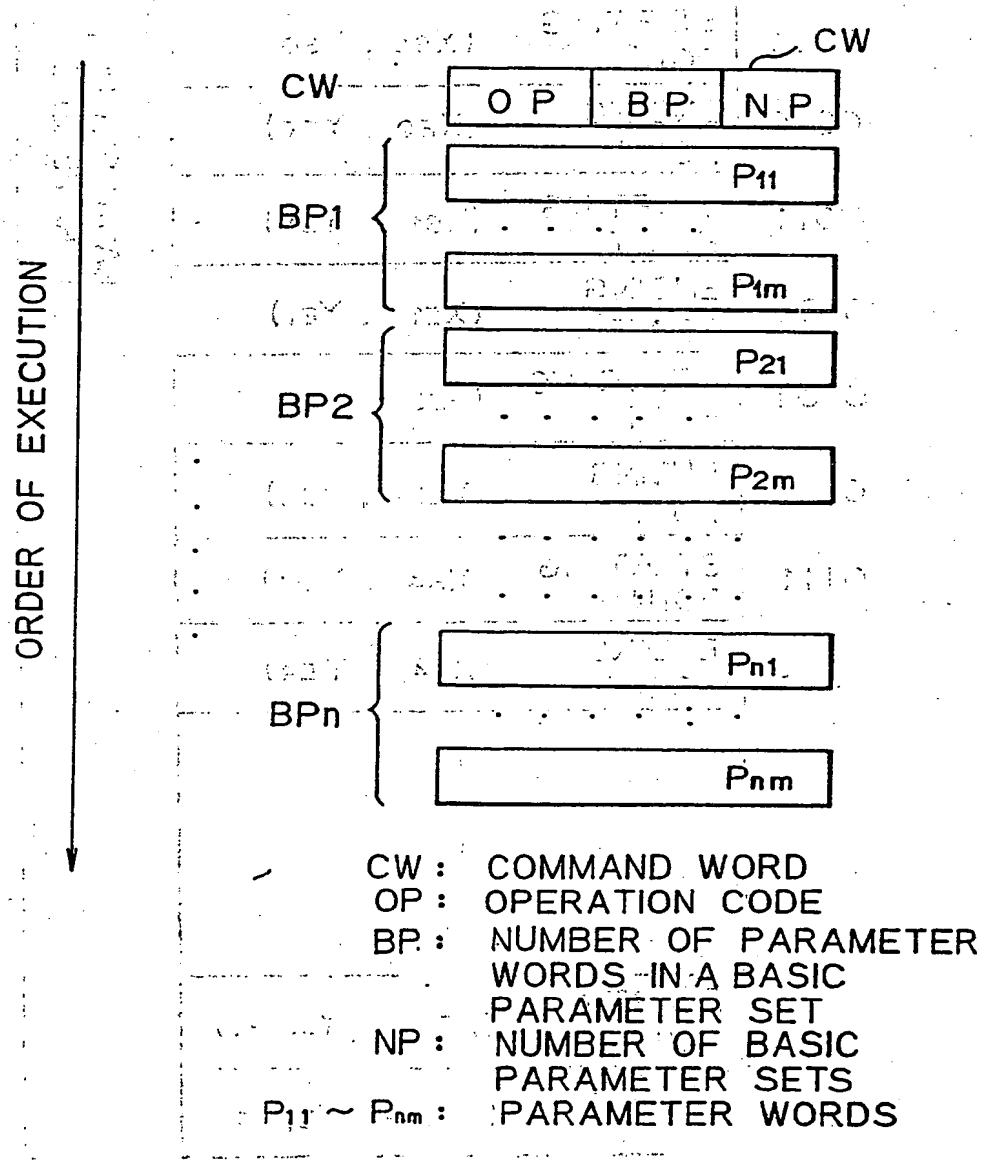

LINE SEGMENT GENERATING COMMAND
COMMAND FETCH
EXECUTION
LINE SEGMENT GENERATING COMMAND
COMMAND FETCH
COMMAND DECORD
PARAMETER FETCH
(Xs1, Ys1)
(Xe1, Ye1)
EXECUTION
LINE SEGMENT GENERATING COMMAND
COMMAND FETCH
COMMAND DECORD
PARAMETER FETCH
(Xs0, Ys0)
(Xe0, Ye0)
EXECUTION
COMMAND FETCH
COMMAND DECORD
PARAMETER FETCH
(Xs2, Ys2)
(Xe2, Ye2)

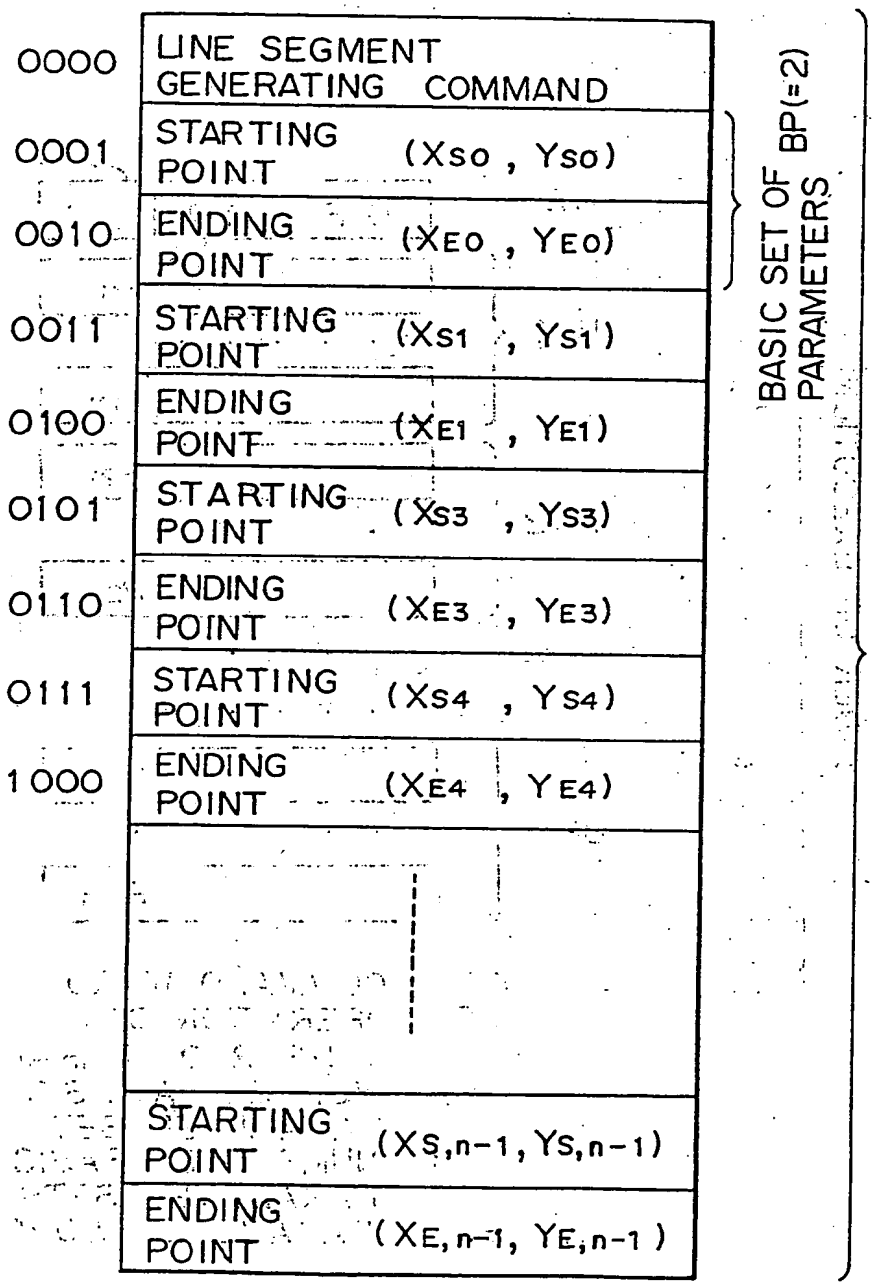
Fig. 6

Fig. 7



BNSDOCID: <EP___0389175A2_I_>

Fig. 8



BNSDOCID: <EP 0389175A2 >

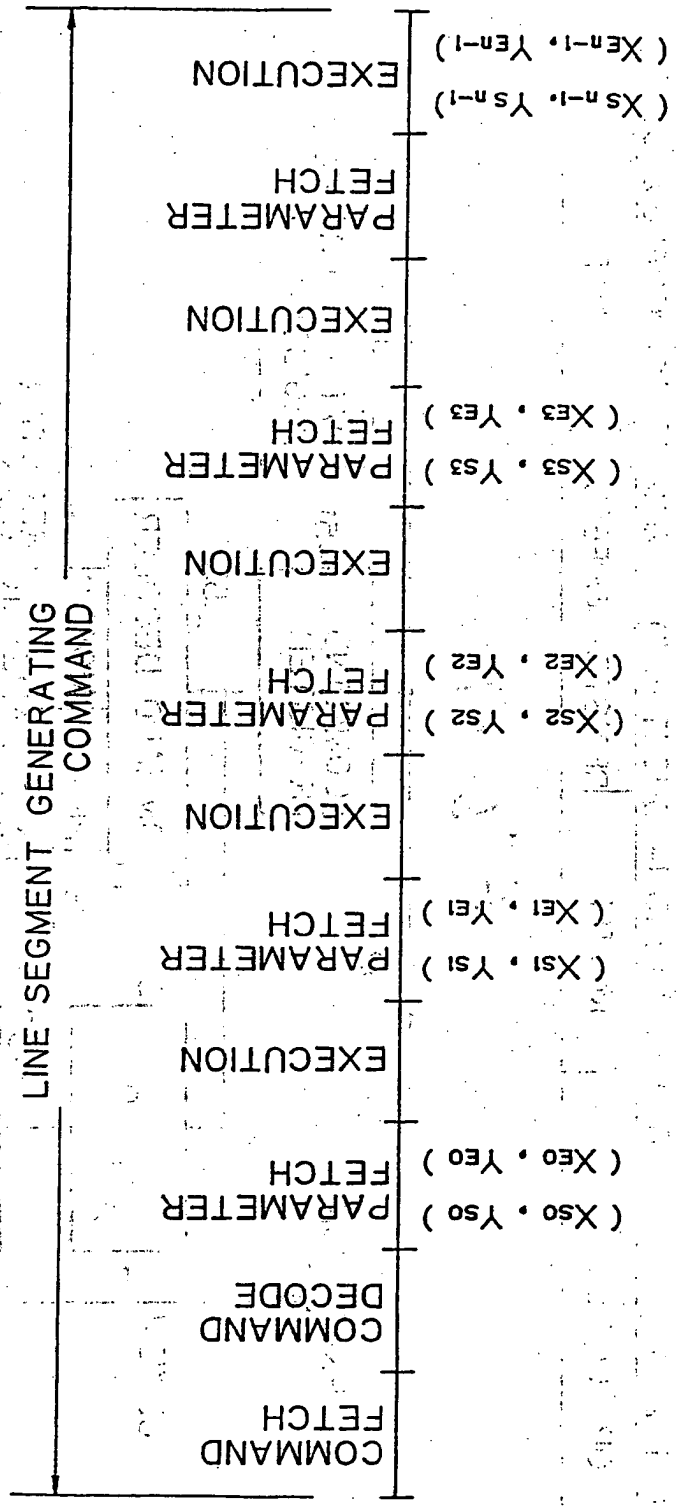

COMMAND FETCH
COMMAND DECODE
PARAMETER FETCH
(Xso , Yso)
(Xeo , Yeo)
EXECUTION
PARAMETER FETCH
(Xs1 , Ys1)
(Xe1 , Ye1)
EXECUTION
PARAMETER FETCH
(Xs2 , Ys2)
(Xe2 , Ye2)
EXECUTION
PARAMETER FETCH
(Xs3 , Ys3)
(Xe3 , Ye3)
EXECUTION
PARAMETER FETCH
EXECUTION
(Xs n-1 , Ys n-1)
(Xe n-1 , Ye n-1)
LINE SEGMENT GENERATING COMMAND

Fig. 9

Fig. 10

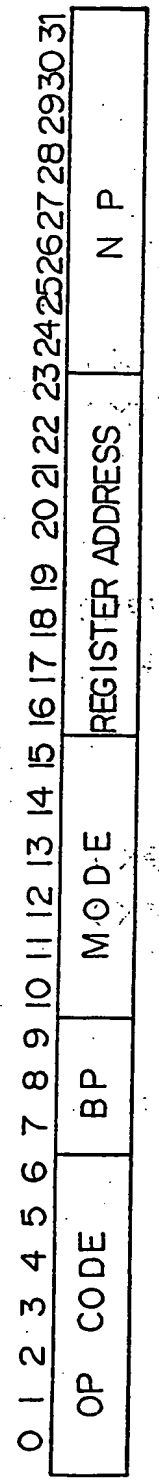
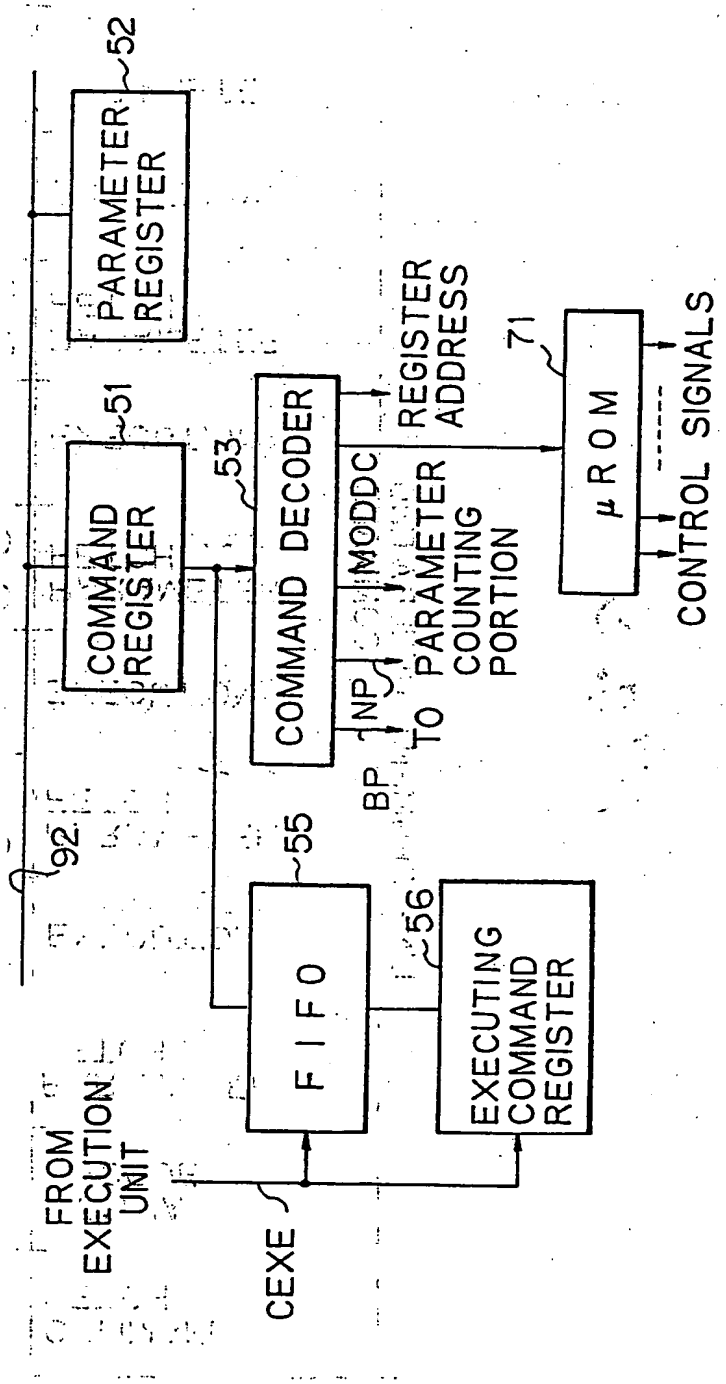
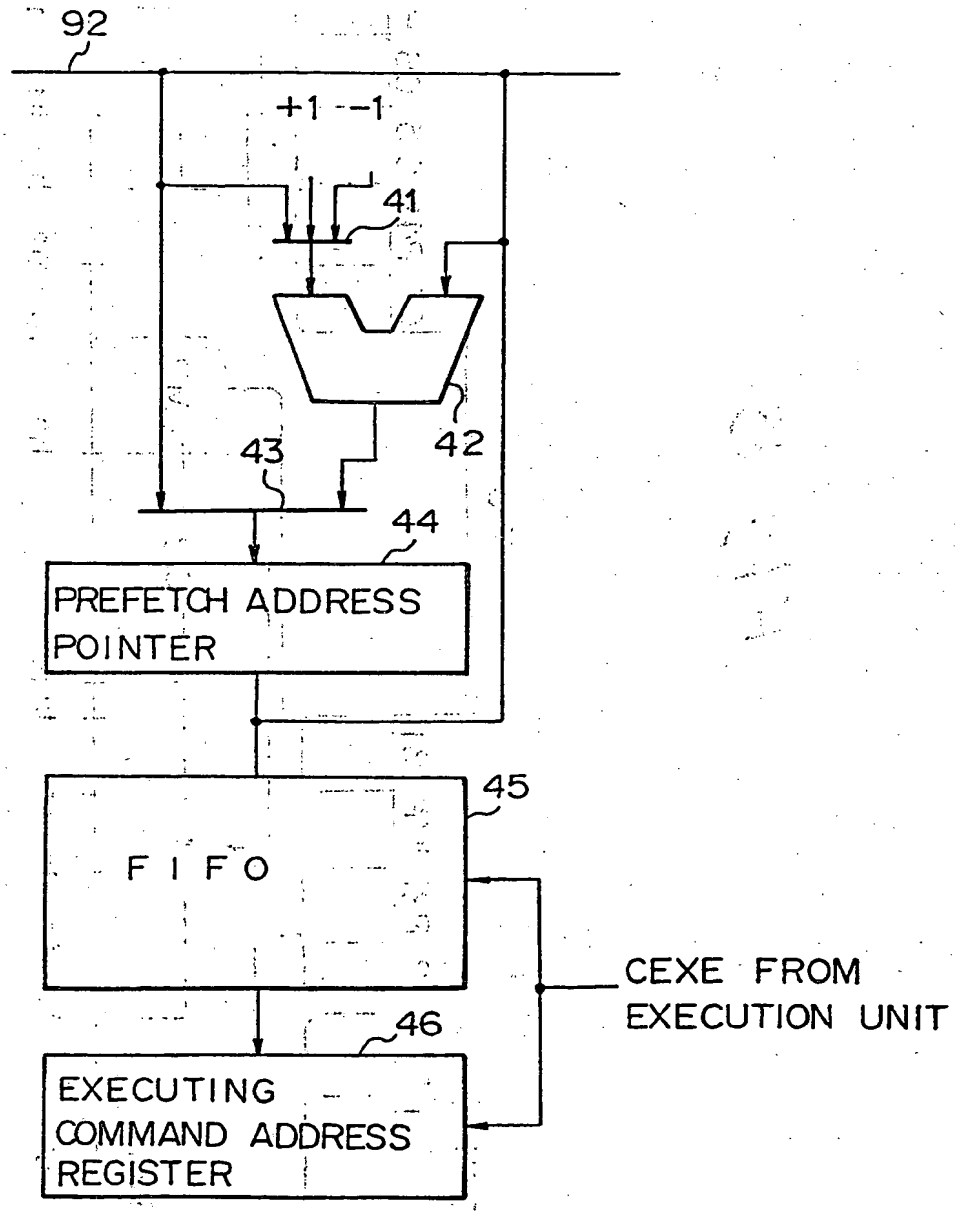


Fig. 11



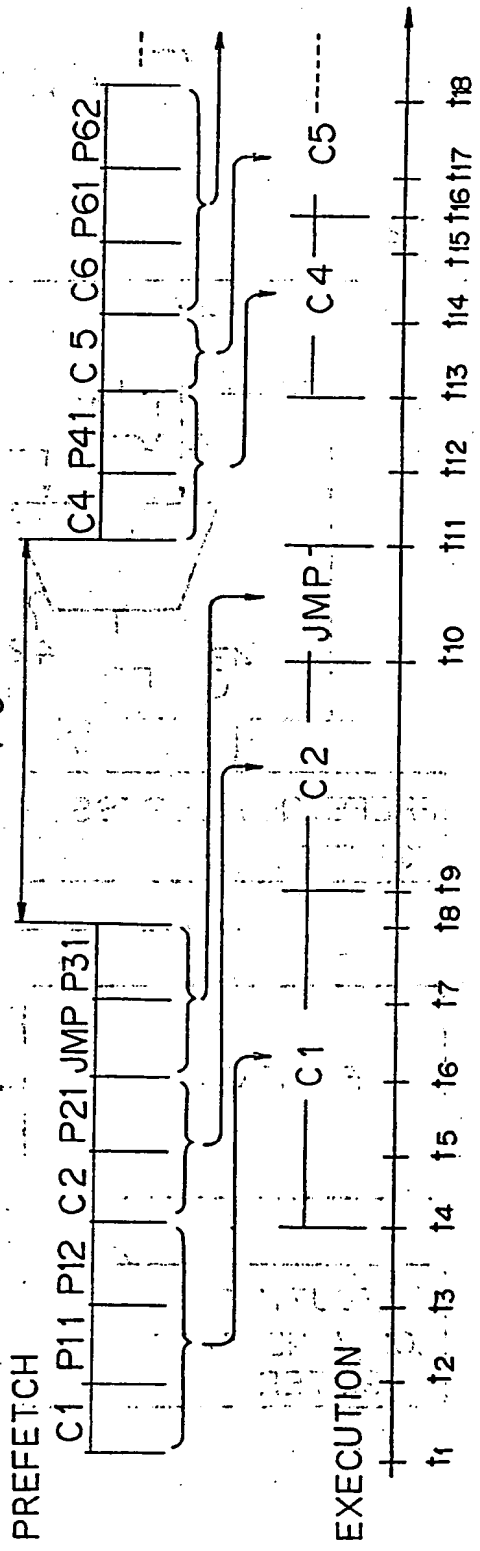
BNSDOCID: <EP 0389175A2 I >

Fig. 12



BNSDOCID: <EP___0389175A2_I_>

Fig. 13


PREFETCH
EXECUTION
C1 P11 P12 C2 P21 JMP P31
C4 P41 C5 C6 P61 P62
T-0
C1
C2
JMP
C4
C5
t1 t2 t3 t4 t5 t6 t7 t8 t9 t10 t11 t12 t13 t14 t15 t16 t17 t18

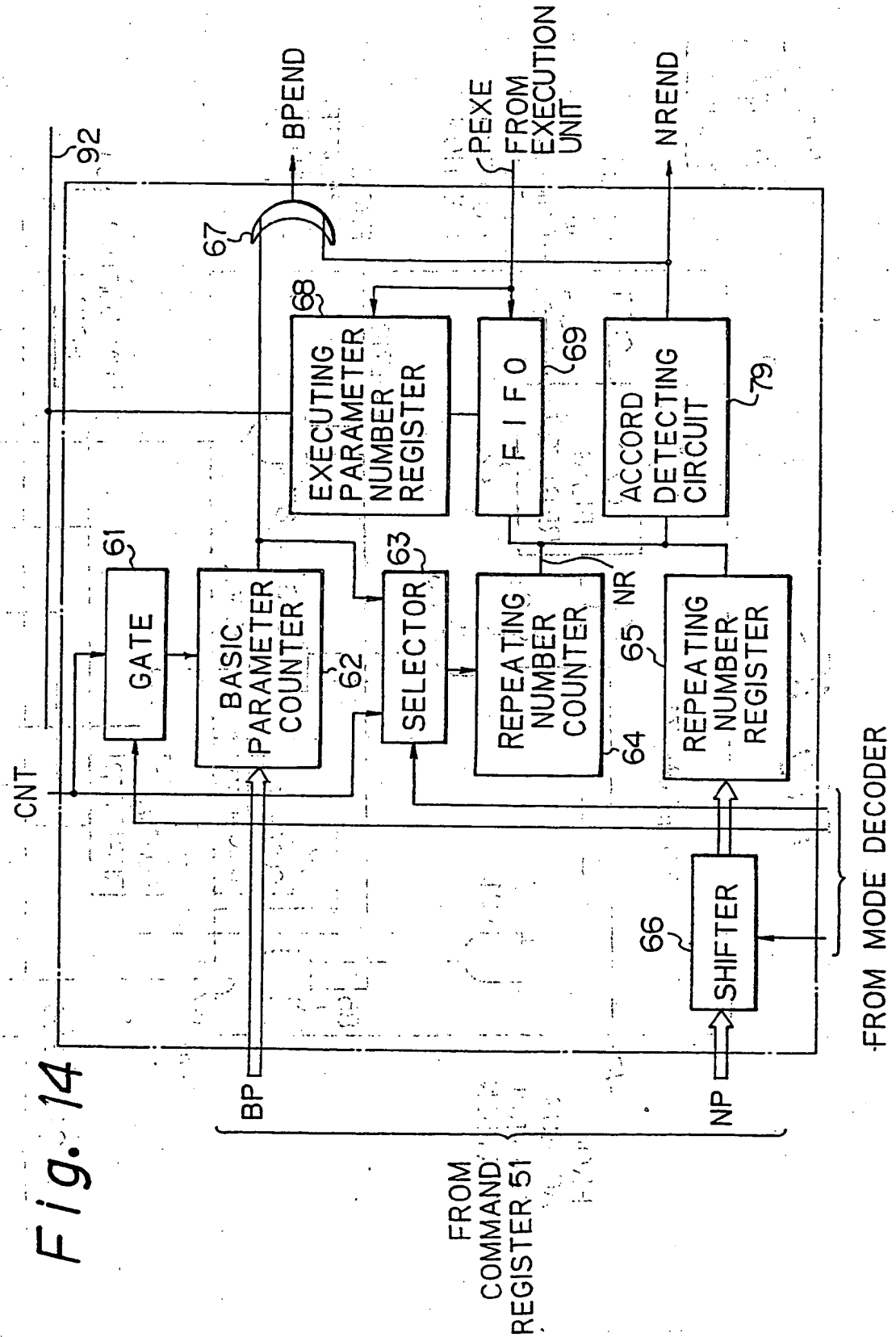

92
BPEND
PEXE FROM EXECUTION UNIT
NREND
67
68
EXECUTING PARAMETER NUMBER REGISTER
F I F O
69
ACCORD DETECTING CIRCUIT
79
61
GATE
BASIC PARAMETER COUNTER
62
63
SELECTOR
REPEATING NUMBER COUNTER
64
NR
65
REPEATING NUMBER REGISTER
CNT
66
SHIFTER
FROM MODE DECODER
BP
NP
FROM COMMAND REGISTER 51

Fig. 14

Fig. 15

Fig. 15A
Fig. 15 B

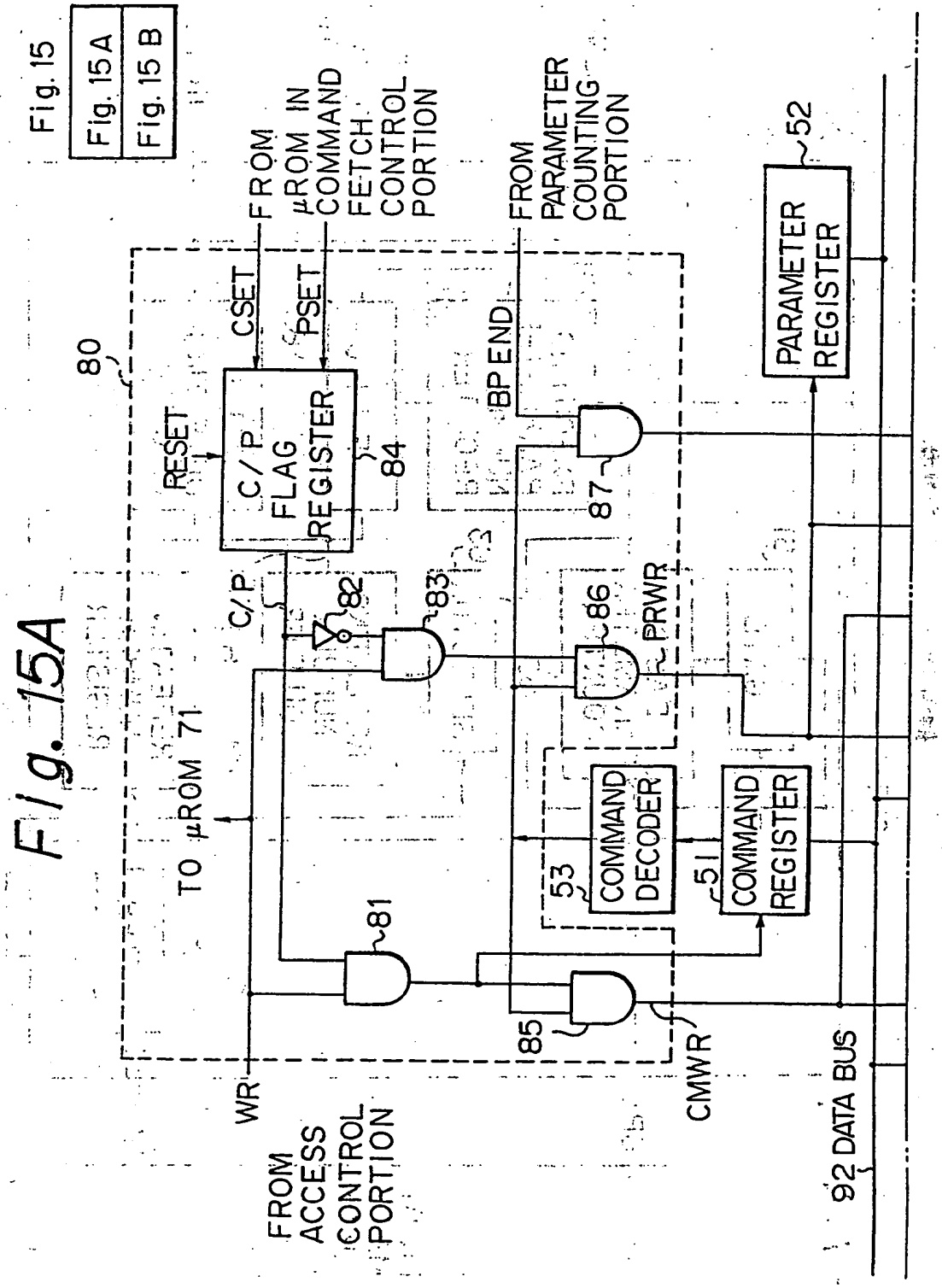
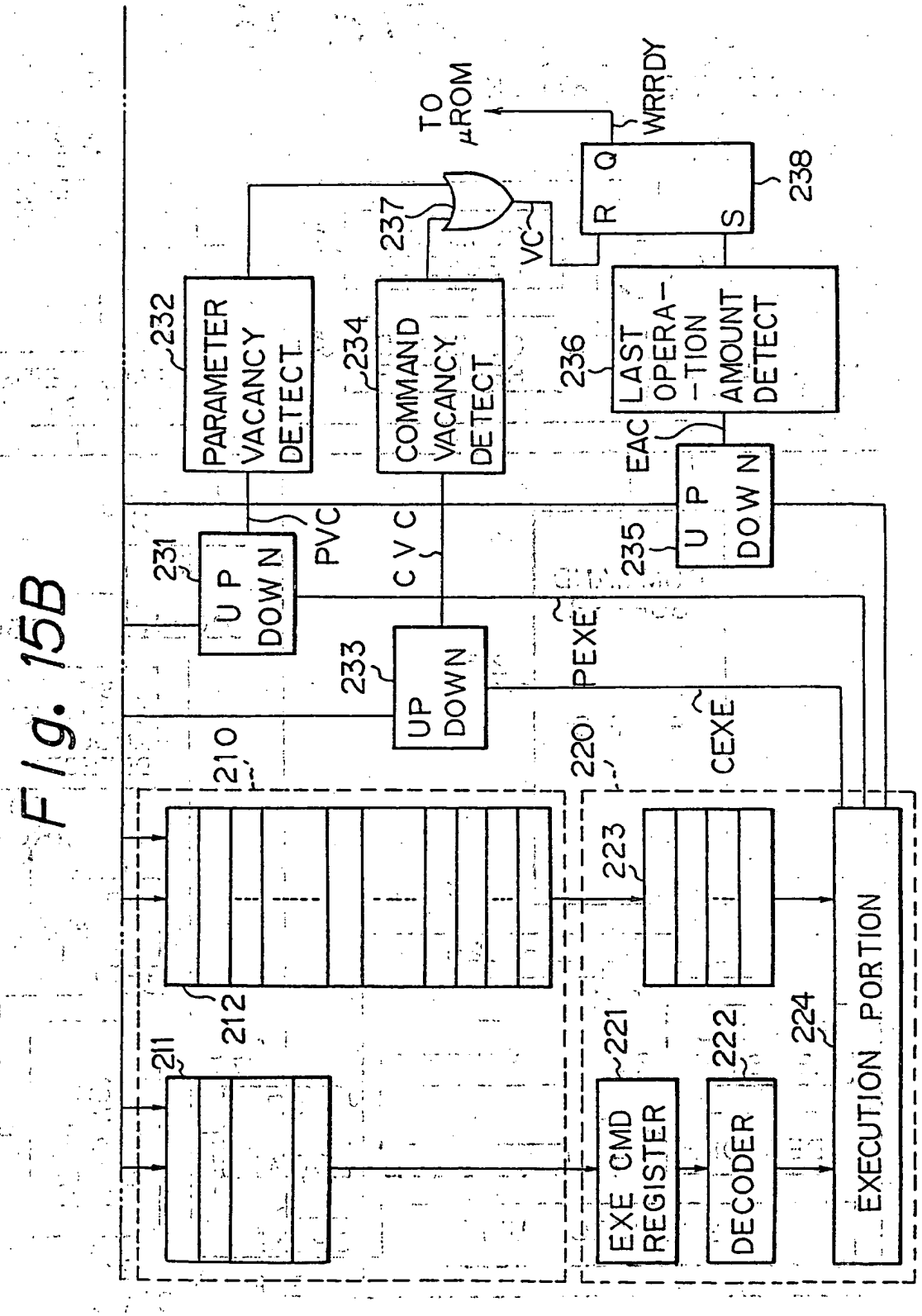


Fig. 15A



F I g. 15B

BNSDOCID: <EP___0389175A2_I_>

Fig. 16

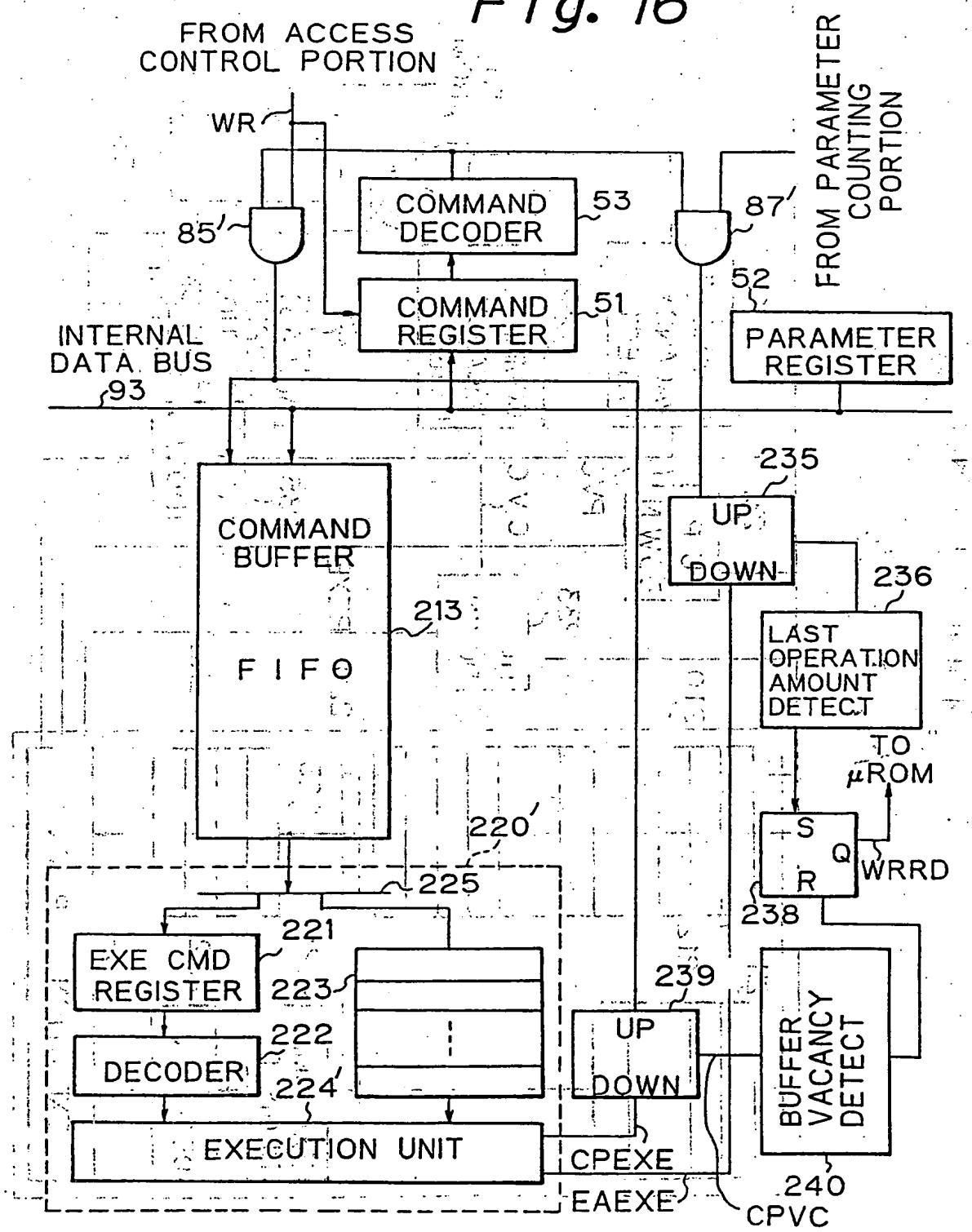
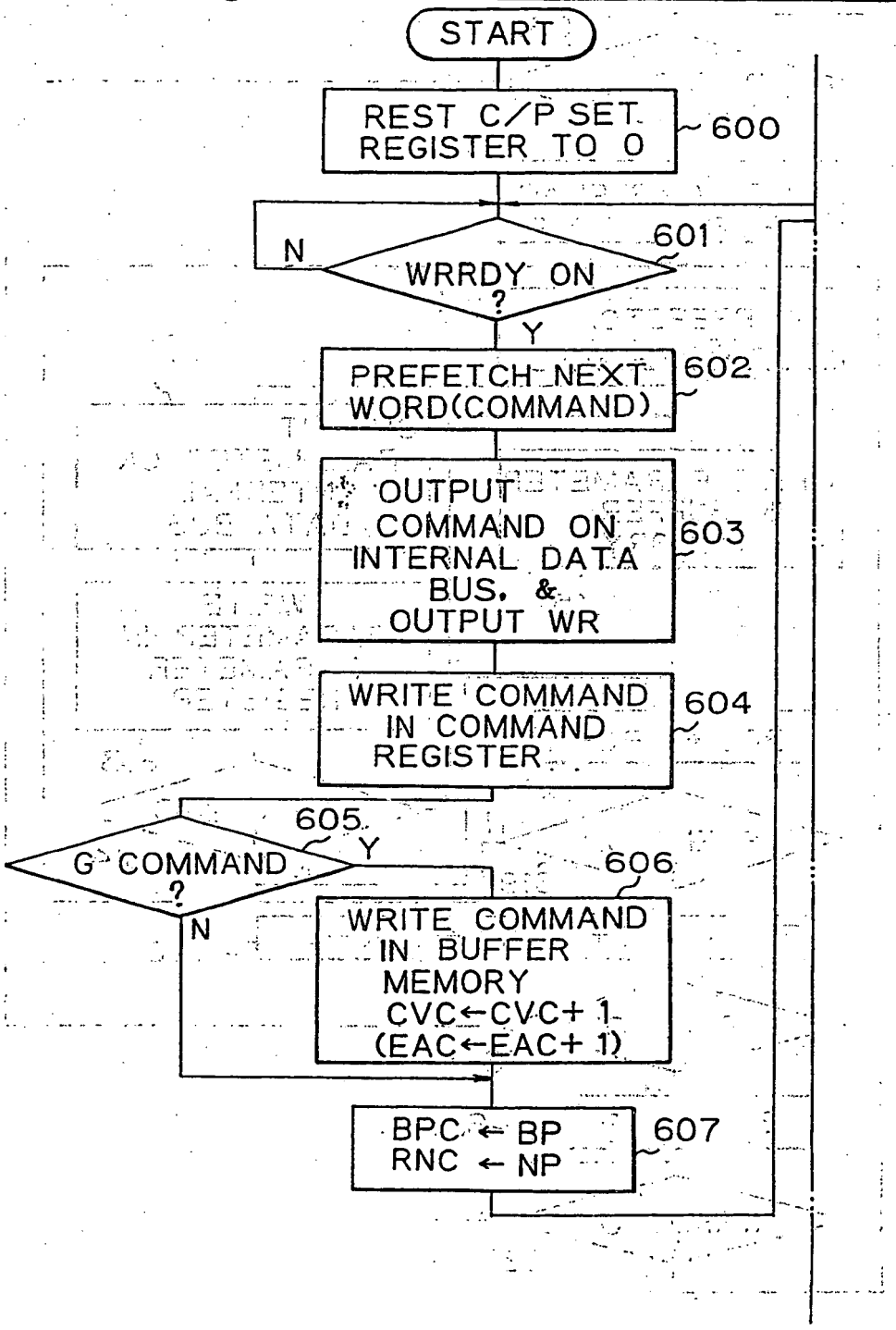

FROM ACCESS CONTROL PORTION
WR
85'
COMMAND DECODER
53
87'
FROM PARAMETER COUNTING PORTION
52
COMMAND REGISTER
51
PARAMETER REGISTER
INTERNAL DATA BUS
93
235
UP
DOWN
236
COMMAND BUFFER
213
FIFO
LAST OPERATION AMOUNT DETECT
TO μROM
220'
S
Q
R
WRRD
238
225
221
EXE CMD REGISTER
223
239
222
DECODER
224'
UP
DOWN
BUFFER VACANCY DETECT
EXECUTION UNIT
CPEXE
EAEXE
240
CPVC

Fig. 17

Fig. 17A

Fig. 17A	Fig. 17B	Fig. 17C



BNSDOCID: <EP___0389175A2_I_>

Fig. 17B

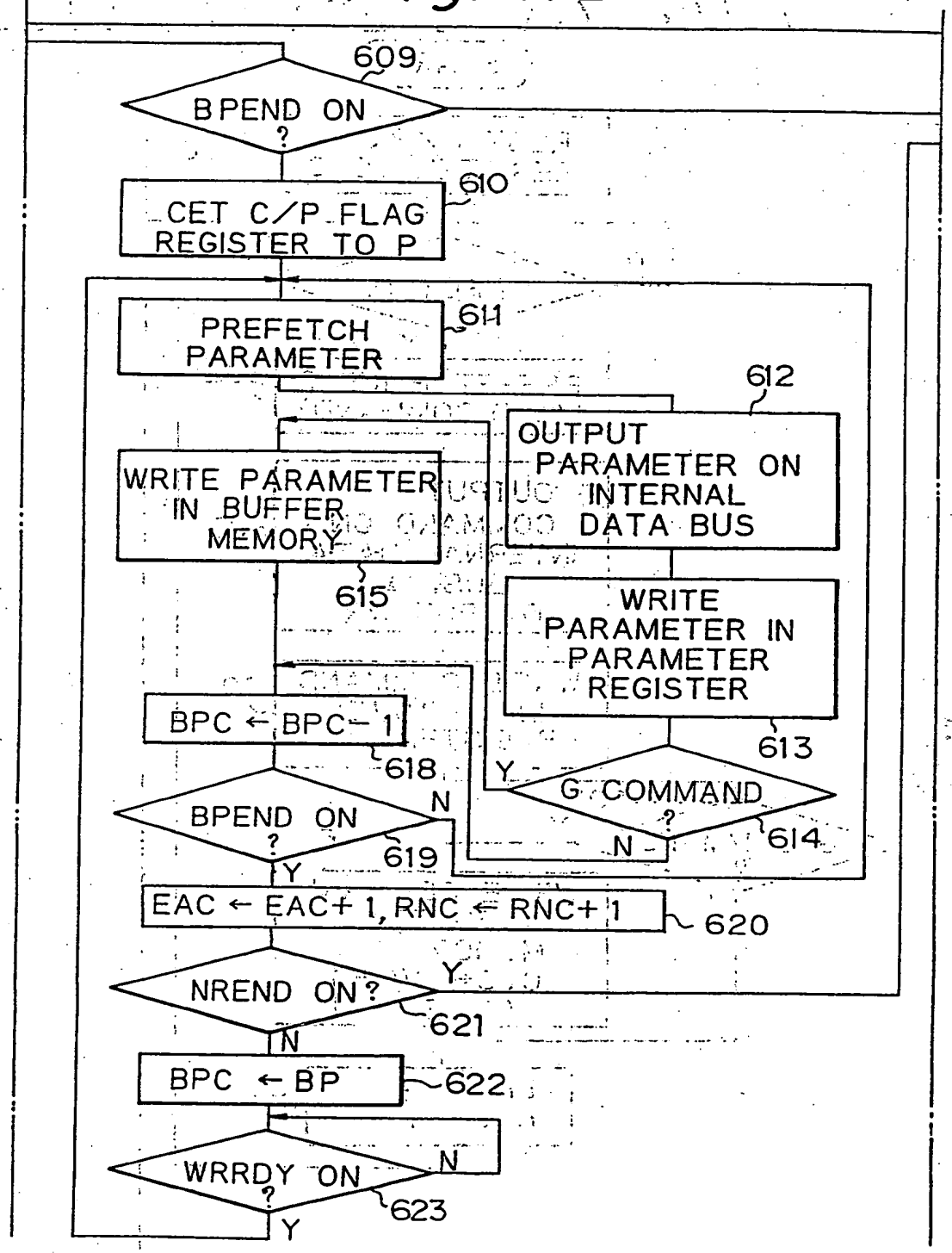
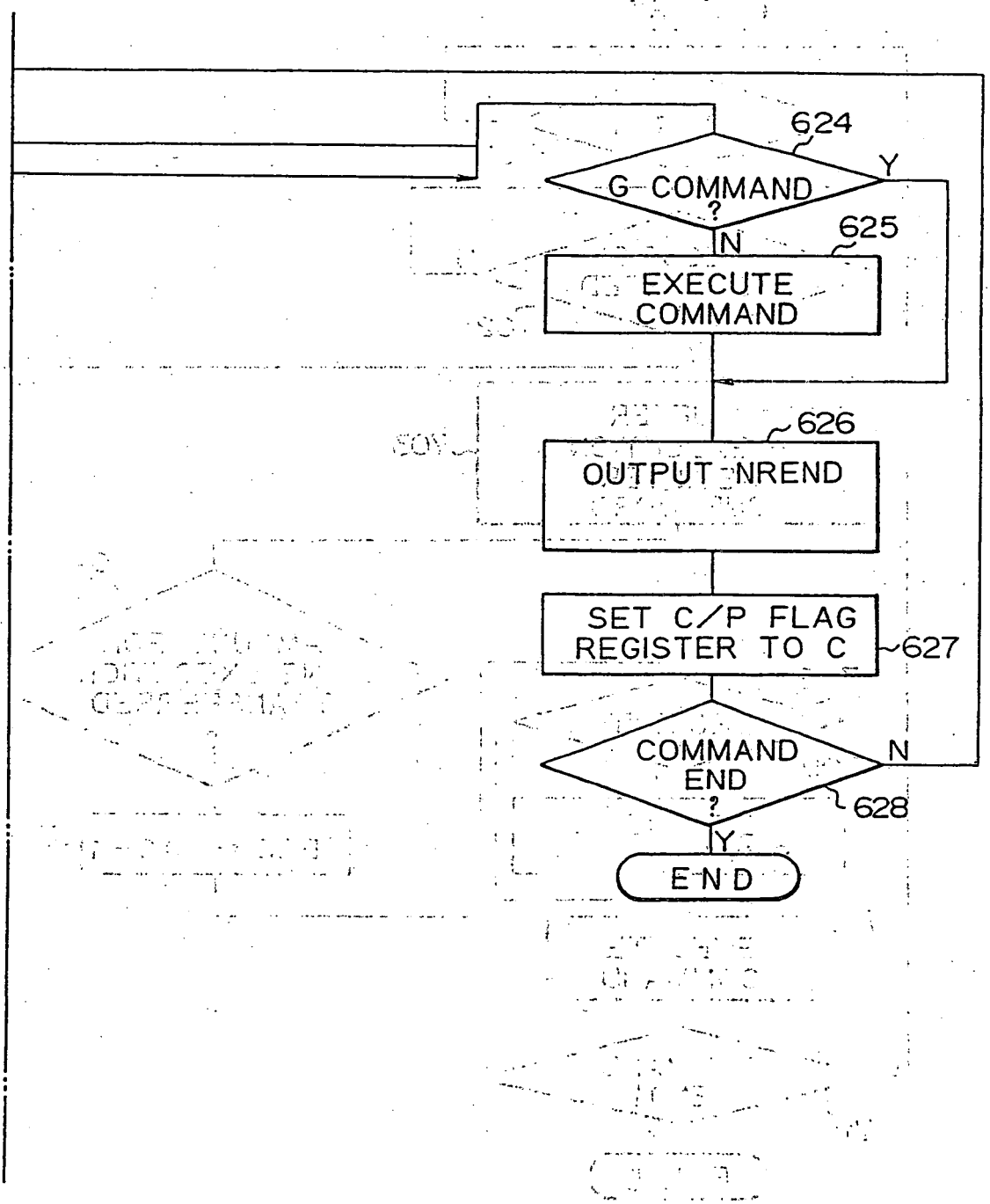

609
BPEND ON ?
610
GET C/P FLAG REGISTER TO P
611
PREFETCH PARAMETER
612
OUTPUT PARAMETER ON INTERNAL DATA BUS
WRITE PARAMETER IN BUFFER MEMORY
615
WRITE PARAMETER IN PARAMETER REGISTER
613
G COMMAND ?
614
Y
N
BPC ← BPC − 1
618
BPEND ON ?
619
N
Y
EAC ← EAC + 1, RNC ← RNC + 1
620
NREND ON ?
621
Y
N
BPC ← BP
622
WRRDY ON ?
623
N
Y

Fig. 17C



BNSDOCID: <EP___0389175A2_I_>

Fig. 18

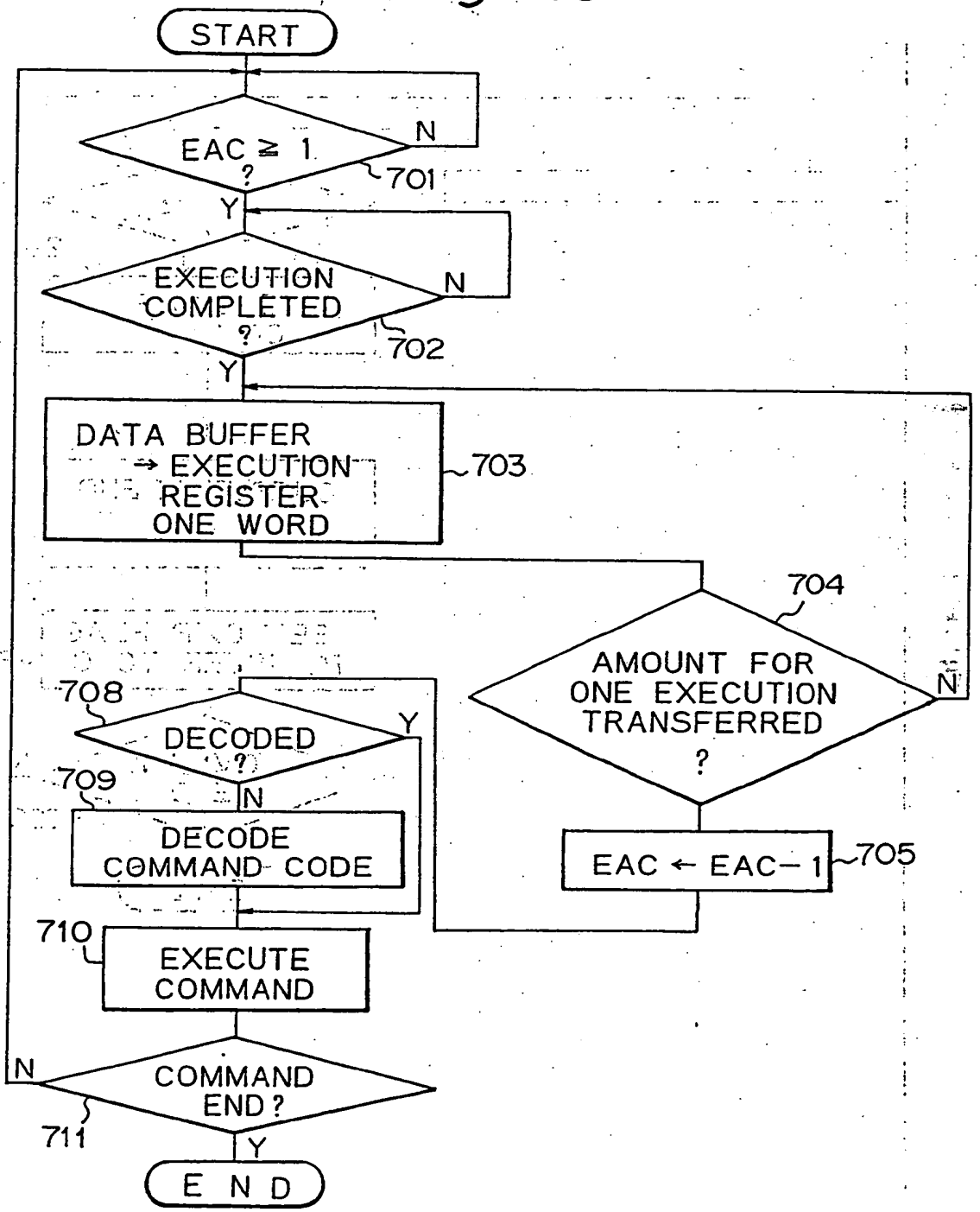
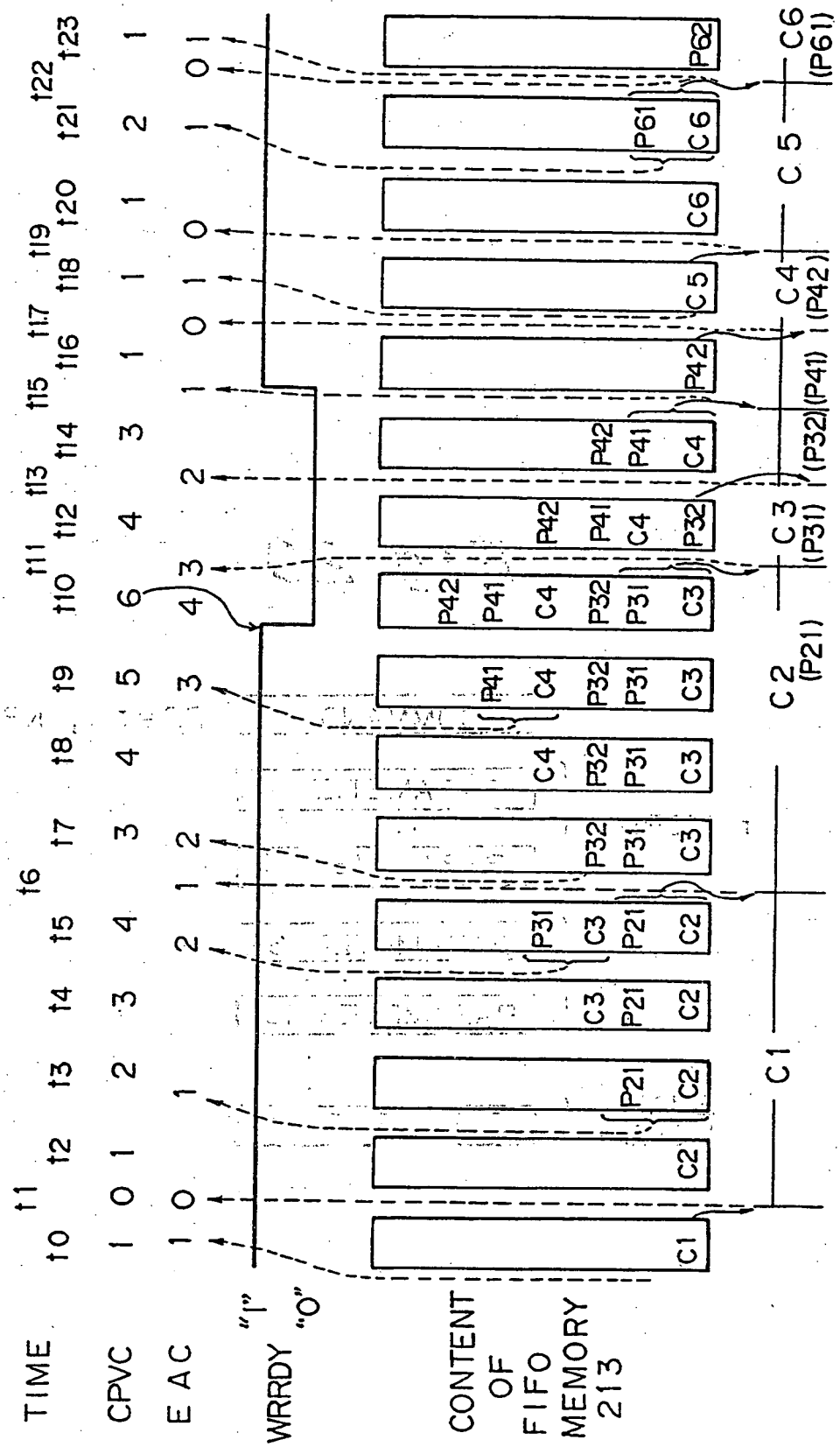

START
EAC ≥ 1 ?
N
701
Y
EXECUTION COMPLETED ?
N
702
Y
DATA BUFFER → EXECUTION REGISTER ONE WORD
703
704
AMOUNT FOR ONE EXECUTION TRANSFERRED ?
N
EAC ← EAC − 1
705
708
DECODED ?
Y
709
N
DECODE COMMAND CODE
710
EXECUTE COMMAND
N
COMMAND END ?
711
Y
E N D

Fig. 19



BNSDOCID: <EP___0389175A2_I_>

Fig. 20

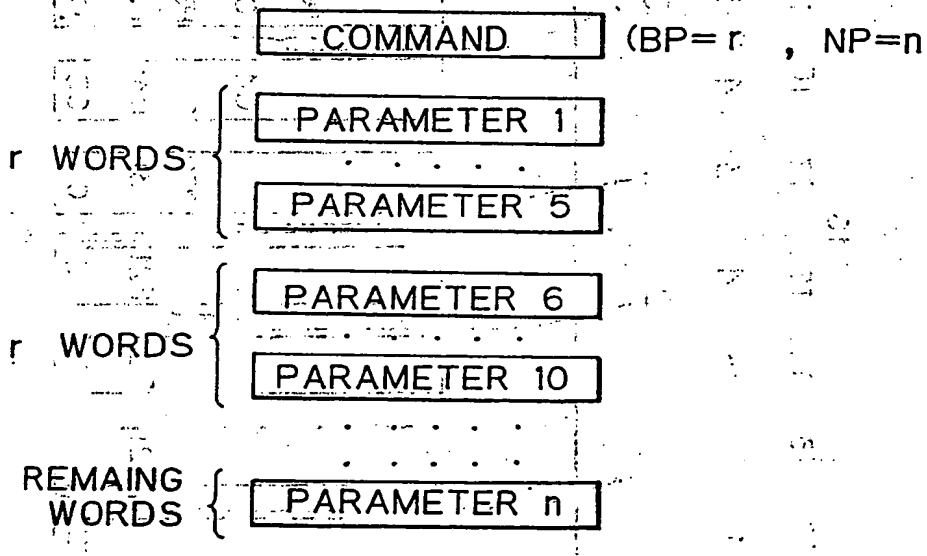

COMMAND
(BP= r , NP=n
r WORDS
PARAMETER 1
PARAMETER 5
r WORDS
PARAMETER 6
PARAMETER 10
REMAING WORDS
PARAMETER n

Fig. 21

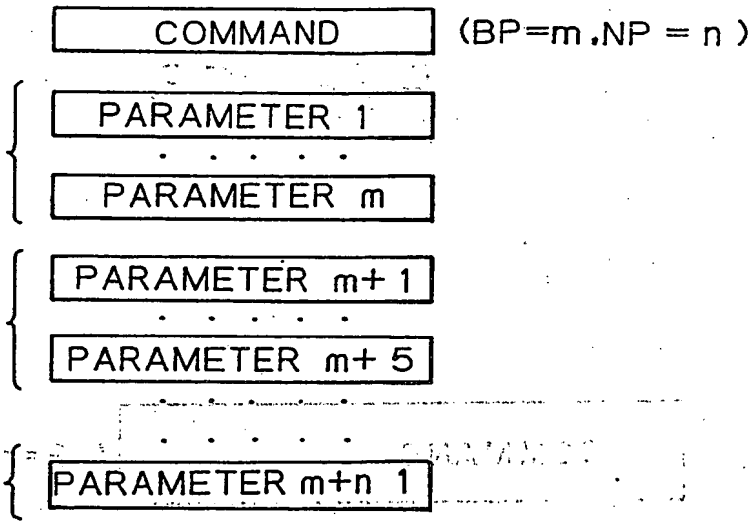
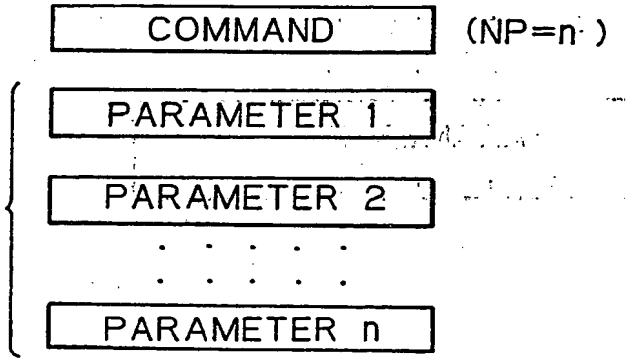
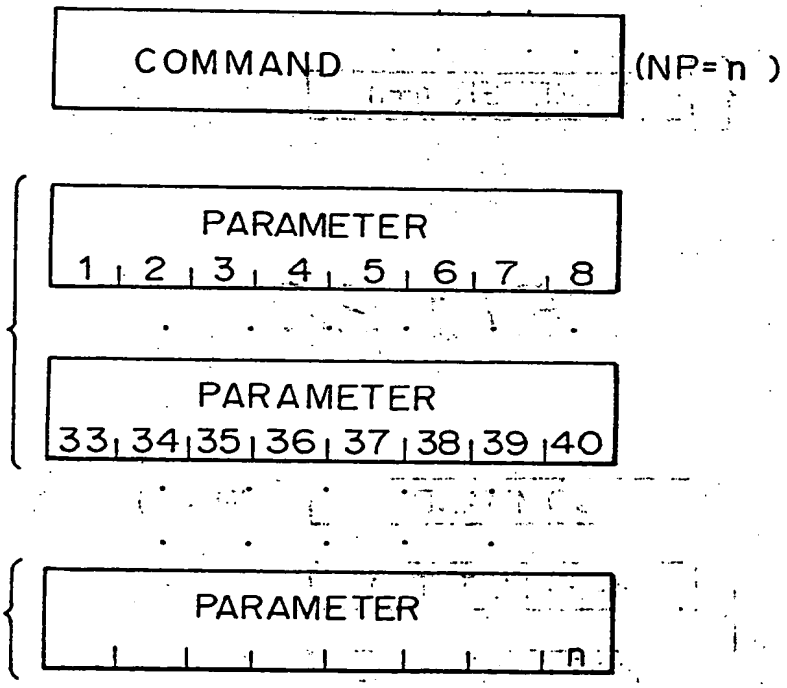


Fig. 22



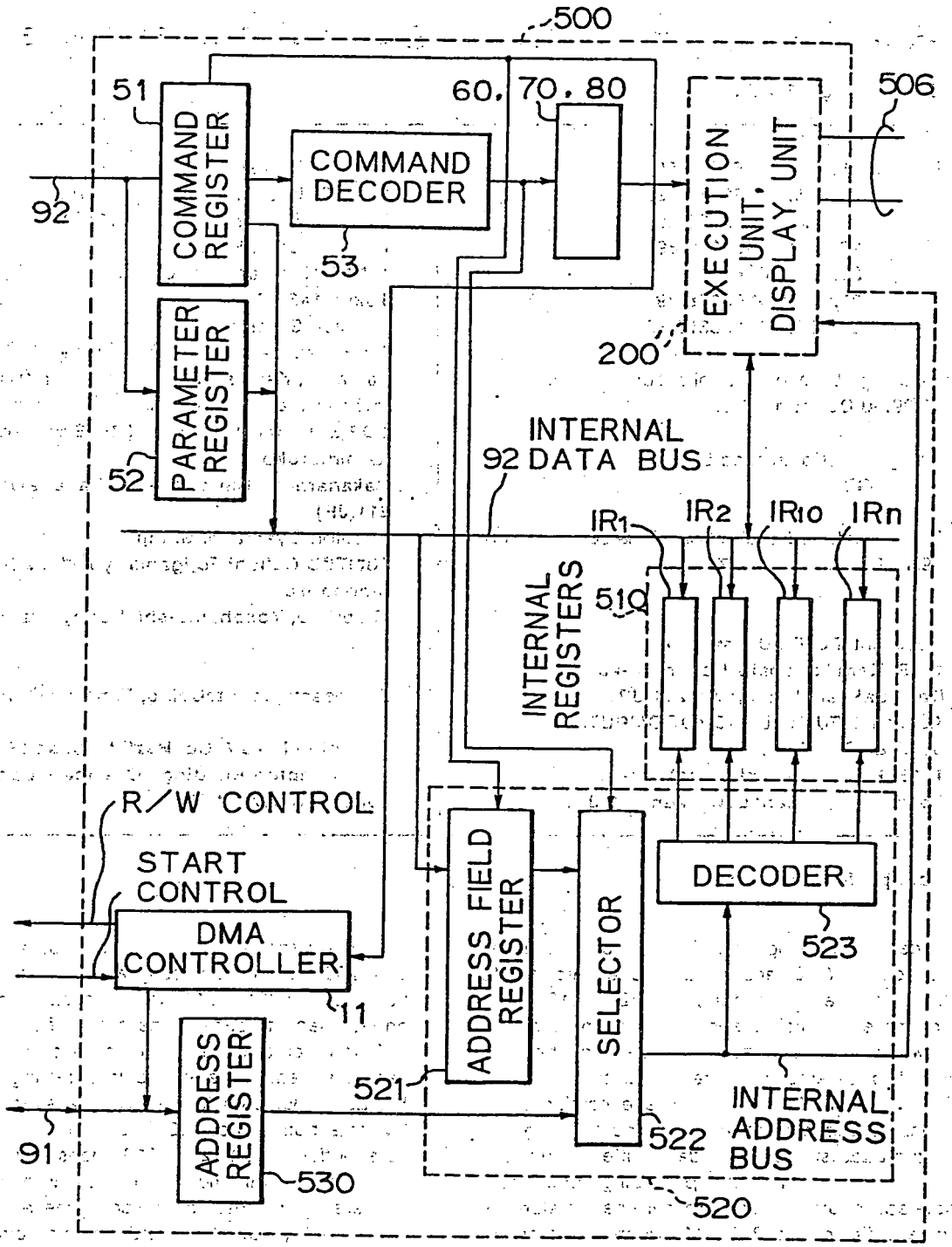
BNSDOCID: <EP___0389175A2_I_>

Fig. 23



RNSDOCID: <EP 0389175A2 I >

Fig. 24



BNSDOCID: <EP___0389175A2_I_>

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 389 175 A3**

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 90302749.8

(51) Int. Cl.5 G06F 15/66, G06F 9/38

(22) Date of filing: 15.03.90

(30) Priority: 15.03.89 JP 62834/89
15.03.89 JP 62832/89
15.03.89 JP 62833/89
16.03.89 JP 62032/89
20.03.89 JP 68377/89
20.03.89 JP 68378/89
26.04.89 JP 108370/89

(43) Date of publication of application:
26.09.90 Bulletin 90/39

(84) Designated Contracting States:
DE FR GB

(88) Date of deferred publication of the search report:
19.11.92 Bulletin 92/47

(71) Applicant: FUJITSU LIMITED
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa 211(JP)
Applicant: FUJITSU MICROCOMPUTER SYSTEMS LIMITED
1015, Kamikodanaka Nakahara-ku
Kawasaki-shi Kanagawa-ken 211(JP)

(72) Inventor: Taniai, Takayoshi
1339, Shimosakunobe, Takatsu-ku
Kawasaki-shi, Kanagawa 213(JP)
Inventor: Sato, Hajime
1-4-12, Kitamagome, Ota-ku
Tokyo 143(JP)
Inventor: Shimura, Hidetoshi
Eri Haitsu A-103, 13-7, Shukugawara 6-chome
Tama-ku, Kawasaki-shi, Kanagawa 214(JP)
Inventor: Saitoh, Tadashi
TOP Shinmaruko 306, 1-1525, Sanno-cho, Kamimaruko
Nakahara-ku, Kawasaki-shi, Kanagawa 211(JP)
Inventor: Oyamada, Shinji
FUJITSU Daiichi Fujigaoki-ryo 211, 23-10, Umegaoka
Midori-ku, Yokohama-shi, Kanagawa 227(JP)

(74) Representative: Stebbing, Timothy Charles et al
Haseltine Lake & Co. Hazlitt House 28
Southampton Buildings Chancery Lane
London WC2A 1AT(GB)

(54) Data prefetch system.

(57) A data processing device (500) contains an execution circuit (200) and a data buffer circuit (210, 213). The above data buffer circuit (210, 213) stores one or more commands and/or one or more parameters which are prefetched, until each of the commands and parameters is read out by the above execution circuit (200). The above execution circuit (200) inputs a command which is the oldest one of the commands stored in the data buffer circuit (210, 213) when an execution of a preceding command is completed, inputs one or more parameters stored in the data buffer circuit (210, 213) when the command input therein requests the parameters, and executes the command input therein, using the parameters when the parameters are input therein. The above data processing device (500) further contains an insufficient space detecting circuit (60, 80, 231 - 235, 237, 238, 239) for detecting a first state of the data buffer circuit (210, 213) in which state there is not enough vacant space in the data buffer circuit (210, 213) in which a further command and/or a parameter can be stored, and a least data detecting circuit (60, 80, 235, 236, 238) for detecting a second state of the data buffer circuit (210, 213) in which state the data buffer circuit (210, 213) does not store data including a command and/or a parameter, which is necessary for a next operation in the above execution circuit (200). A prefetch control circuit (100), provided in either in the device (500) or outside of the device, starts the prefetch operation and continues successive prefetch operations when the second state is detected, and stops the prefetch operation

EP 0 389 175 A3

when the first state is detected. The system using the abovo device (500) contains a memory (502) which stores the above commands and parameters in the order of execution in the execution circuit (200).

Fig. 3A

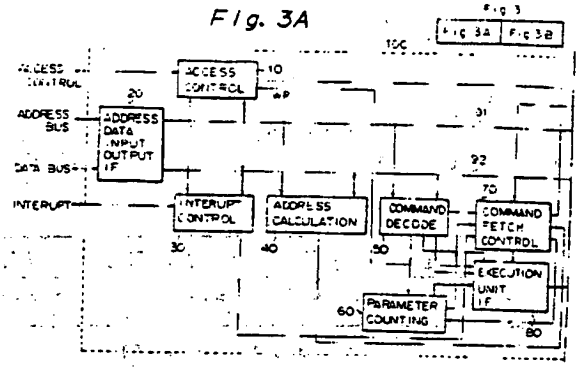
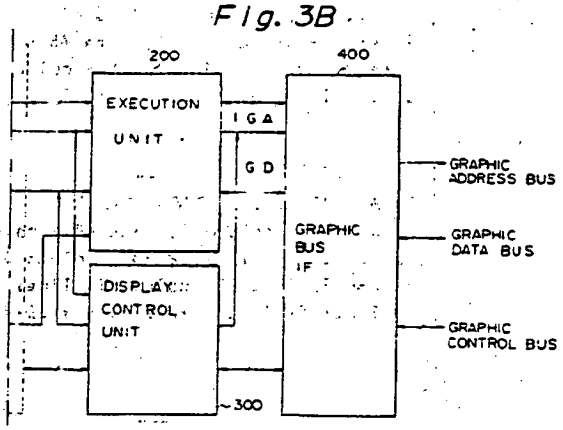


Fig. 3B



BNSDOCID: <EP___0389175A3_I_>

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 90 30 2749

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
X	US-A-4 648 034 (HENINGER)	1,2,30, 32,33	G06F15/66 G06F9/38
Y	* abstract; column 1, lines 40-63; column 3, lines 25-36; column 12, line 45 - column 13, line 45; column 14, line 5 - column 15, line 25; column 15, line 43 - column 16, line 64 *	3-5,14, 16-19, 21,28, 31,34, 36,44, 47-49,56	
Y	EP-A-0 261 685 (HITACHI LTD.)	5,14	
Y	* abstract; column 1, line 49 - column 3, line 20; column 4, lines 23-34; column 5, line 29 - column 6, line 10; column 6, lines 36-52; column 8, lines 5-16; column 9, lines 29-47 *	16-18, 21,28, 30-32, 36,47, 49,56	
A		6,22,37, 50	
Y	IBM TECHNICAL DISCLOSURE BULLETIN. vol. 25, no. 9, February 1983, NEW YORK US pages 4722 - 4727; VENESKI AND WYATT: 'Asynchronous pipelined instruction prefetch mechanism' * the whole document *	3,4,19, 30,32, 34,48	TECHNICAL FIELDS SEARCHED (Int. Cl.5) G06F
Y	IEEE MICRO. vol. 8, no. 3, June 1988, NEW YORK US pages 77 - 87; S. IACOBOVICI: 'A pipelined interface for high floating-point performance with precise exceptions'	44	
A	* page 79, left column, lines 38-46; page 80, left column, line 30 - page 82, left column, line 16; page 87, left column, lines 21-38 *	13,27,55	
A	EP-A-0 079 393 (FANUC LTD.) * abstract; page 6, line 23 - page 7, line 27; page 11, line 3 - page 13, line 17; figure 4 *	18,32,47	

-/--

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	14 SEPTEMBER 1992	WEINBERG L.F.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0001)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number

EP 90 30 2749

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int. Cl.5)
A	US-A-4 763 242 (LEE ET AL.) 9 August 1988 * column 2, lines 40-56; column 4, lines 48-63; column 7, lines 1-12 * ---	12,13,26 27,44, 54,55	
A	EP-A-0 284 364 (SEIKO INSTRUMENTS INC.) 28 September 1988 * column 2, line 30 - column 3, line 4; column 4, lines 31-42; column 8, line 27 - column 9, line 21 * -----	1,3,19, 21,30, 32,33,49	

TECHNICAL FIELDS SEARCHED (Int. Cl.5)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	04 SEPTEMBER 1992	WEINBERG L.F.

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P0401)

BNSDOCID: <EP___0389175A3_I >